

**WO 2005/071549 A1**



LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NA,  
NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE,  
SG, SK, SL, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US,  
UZ, VC, VN, YU, ZA, ZM, ZW.

IE, IS, IT, LT, LU, MC, NL, PL, PT, RO, SE, SI, SK, TR),  
OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML,  
MR, NE, SN, TD, TG).

添付公開書類:

— 国際調査報告書

- (84) 指定国 (表示のない限り、全ての種類の広域保護が可能): ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), ユーラシア (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), ヨーロッパ (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU,

2文字コード及び他の略語については、定期発行される各PCTガゼットの巻頭に掲載されている「コードと略語のガイダンスノート」を参照。

アクセスを行うファイルシステムインターフェース制御部120を設ける。これにより、アクセス装置100が半導体メモリ装置110の特性を意識することなく、ファイルシステムインターフェース制御部120を介して最適なファイルアクセスを行うことができる。

## 明 細 書

### 半導体メモリ装置及びその制御方法

#### 技術分野

- [0001] 本発明は、不揮発性メモリに格納したデータをファイルシステムにより管理する半導体メモリ装置及び半導体メモリ装置の制御方法に関する。

#### 背景技術

- [0002] 音楽コンテンツや、映像データ等のデジタルデータを記録する記録媒体には、磁気ディスク、光ディスク、光磁気ディスク、半導体メモリ装置等、様々な種類の記録媒体が存在する。半導体メモリ装置は、例えばカード形に構成され、記録素子としてフラッシュROM等の半導体不揮発性メモリを使用している。以下半導体メモリカードとして説明するが、他の形状の半導体メモリ装置にも本発明が適用される。半導体メモリカードは記録媒体の小型化が図れることから、デジタルスチルカメラや携帯電話端末等、小型の携帯機器を中心に急速に普及しつつある。
- [0003] 半導体メモリカードに格納されたデータは、ファイルシステムにより管理されており、ユーザは格納されたデータをファイルとして容易に取り扱うことができる。従来使用されているファイルシステムとして、FATファイルシステム(非特許文献1を参照)がある。又UDFファイルシステム(Universal Disk Format)(非特許文献2を参照)があり、更にNTFSファイルシステム(New Technology File System)等が存在する。これらファイルシステムによりデータが管理された半導体メモリカードは、同一のファイルシステムを解釈する機器間でファイルを共有することができる。このため、機器間でデータを授受することが可能となる。
- [0004] ファイルシステムでは、データを記録する情報記録領域を、最小アクセス単位であるセクタ、及びセクタの集合であるクラスタに分割して管理し、1つ以上のクラスタをファイルとして管理する。ファイルのデータが格納される領域は、空き領域からクラスタ単位で割り当てられ、1つのファイルのデータが必ずしも連続領域に格納されるとは限らない。連続領域に格納されていないファイルのデータを読み書きする場合、読み書きの間にシーク動作が発生するため、連続領域に格納されたファイルのデータに

比べ、読み書き速度が低下するという問題がある。

- [0005] 従来、このような問題を解決する方法として、画像処理装置において原稿1頁分のデータが連続領域に格納されるように、データ書き込みを制御する方法が例えば特許文献1に提案されている。この従来の方法では、データ書き込み時に固定長の連続領域に必ずデータを書き込むことで、データ読み出し時に一定の処理時間で処理が完了できることを保証している。

特許文献1:特開2002-29101号公報

非特許文献1:ISO/IEC9293、“Information technology—Volume and file structure of disk cartridges for information”、1994年

非特許文献2:OSTA Universal Disk Format Specification Revision 1.50、1997年

#### 発明の開示

#### 発明が解決しようとする課題

- [0006] しかしながら、上記の従来技術には次のような問題点がある。従来の制御方法では、画像処理装置の処理単位である原稿1頁分のデータサイズを連続領域の単位として使用している。すなわち、アプリケーションが取り扱うデータに適したサイズを基に連続領域の単位を決定している。この方法は、記録媒体への書き込み単位の違いにより書き込み速度に差が生じない記録媒体には有効である。しかしながら、半導体メモリカードでは、書き込み単位が書き込み速度に大きく影響し、かつ書き込み単位と書き込み速度の関係は、使用する半導体メモリの特性や管理方法により異なる。このため、全ての半導体メモリカードにおいて最適なアクセス方法は一意に定まらず、従来例のようにデータサイズを固定しても、全ての半導体メモリカードに対し高速にアクセスすることはできない。

- [0007] 本発明では上記問題点に鑑み、半導体メモリカード内に半導体メモリカードの特性に関する情報を格納し、その情報を基に半導体メモリカードの特性に適したファイルアクセスを行うファイルシステムインターフェース制御部を半導体メモリカード内に備えることにより、アクセス装置が半導体メモリカードの特性を意識することなく、最適なファイルアクセスを行えるようにすることを目的とする。

### 課題を解決するための手段

[0008] 本発明による半導体メモリ装置は、複数のセクタからなり、前記複数のセクタのうち連続する特定個のセクタがデータ消去の最小単位であるブロックとしてグループ化され、外部のアクセス装置から送られたデータを書き込み、及び読み出す不揮発性メモリと、前記アクセス装置から制御信号を含むコマンドが入力されたとき、前記不揮発性メモリに対するデータの消去、書き込み、及び読み出しを制御するメモリコントローラと、前記不揮発性メモリの消去ブロックサイズを含む半導体メモリ装置の物理的特性に関する装置情報を格納する装置情報格納部と、前記装置情報格納部に格納された装置情報に基づいて前記不揮発性メモリに対してファイルアクセス処理を実施するファイルシステムインターフェース制御部と、を具備するものである。

[0009] 本発明による半導体メモリ装置の制御方法は、複数のセクタからなり、前記複数のセクタのうち連続する特定個のセクタがデータ消去の最小単位であるブロックとしてグループ化された不揮発性メモリを有する半導体メモリ装置において、装置情報格納部に前記不揮発性メモリの消去ブロックサイズを含む半導体メモリ装置の物理的特性に関する装置情報をあらかじめ格納し、外部のアクセス装置より制御信号を含むコマンドを受付け、ファイルシステムインターフェース制御部により前記装置情報格納部に格納された装置情報に基づいて前記不揮発性メモリ内に対してファイルアクセス処理を実施し、受付けたコマンドに基づいて前記不揮発性メモリに対するデータの消去、書き込み、及び読み出しを行うものである。

### 発明の効果

[0010] 本発明によれば、格納データをファイルシステムより管理している半導体メモリ装置において、半導体メモリ装置の特性に関する情報を格納するメモリ装置情報格納部と、そのメモリ装置情報を基に半導体メモリ装置の特性に適したファイルアクセスを行うファイルシステムインターフェース制御部とを半導体メモリ装置内に備えることにより、アクセス装置が半導体メモリ装置の特性を意識することなく、最適なファイルアクセスを行うことが可能となる。

### 図面の簡単な説明

[0011] [図1]本発明の実施の形態1における半導体メモリカード及びアクセス装置の構成図

である。

[図2]実施の形態1における消去ブロックとセクタとの関係例を示した説明図である。

[図3]実施の形態1における半導体メモリカードへの消去ブロック倍数長のデータ書き込み処理を示したフローチャートである。

[図4]実施の形態1における半導体メモリカードへの1セクタ分のデータ書き込み処理を示したフローチャートである。

[図5]本発明の各実施の形態に用いられるFATファイルシステムの構成図である。

[図6]実施の形態1におけるFATファイルシステムのデータ書き込み処理を示したフローチャートである。

[図7]実施の形態1におけるFATファイルシステムのデータ書き込み前の状態を示した説明図である。

[図8]実施の形態1におけるFATファイルシステムのデータ書き込み後の状態を示した説明図である。

[図9]実施の形態1におけるファイルシステムAPIの一覧を示した説明図である。

[図10]実施の形態1におけるアクセス装置の内部処理を示したフローチャートである。

[図11]実施の形態1における半導体メモリカード内部のカード種別取得コマンド処理を示したフローチャートである。

[図12]実施の形態1における半導体メモリカード内部のOPENコマンド処理を示したフローチャートである。

[図13]実施の形態1における半導体メモリカード内部のWRITEコマンド処理を示したフローチャートである。

[図14]実施の形態1における半導体メモリカード内部のCLOSEコマンド処理を示したフローチャートである。

[図15]実施の形態1におけるデータ配置例を示した説明図である。

[図16]本発明の実施の形態2(その1)における半導体メモリカード及びアクセス装置の構成図である。

[図17]本発明の実施の形態2(その2)における半導体メモリカード及びアクセス装置

の構成図である。

[図18]実施の形態2における低レベルIOAPIの一覧を示した説明図である。

[図19]実施の形態2におけるアクセス装置の内部処理を示したフローチャートである。

[図20]実施の形態2におけるアクセス装置内部のファイルシステム制御部におけるファイルオープン処理を示したフローチャートである。

[図21]実施の形態2におけるアクセス装置内部のファイルシステム制御部におけるファイルデータ書き込み処理を示したフローチャートである。

[図22]実施の形態2におけるアクセス装置内部のファイルシステム制御部におけるファイルクローズ処理を示したフローチャートである。

[図23]本発明の実施の形態3における半導体メモ리카ード及びアクセス装置の構成図である。

[図24]実施の形態3におけるフォーマット後のファイルシステムの構成例を示した説明図である。

[図25]本発明の実施の形態4における半導体メモ리카ード及びアクセス装置の構成図である。

[図26]実施の形態4における半導体メモ리카ード内部のRAMにおけるデータ格納例を示した説明図である。

[図27]実施の形態4におけるキャッシュ管理テーブルの一例を示した説明図である。

[図28]実施の形態4における半導体メモ리카ード内部の同期制御部における処理手順を示したフローチャートである。

[図29]本発明の実施の形態5における半導体メモ리카ード及びアクセス装置の構成図である。

[図30]実施の形態5における半導体メモ리카ード内部処理を示したフローチャートである。

#### 符号の説明

- [0012] 100 アクセス装置  
101, 112 CPU

102, 113 RAM  
103 スロット  
104, 117 ROM  
105 アプリケーションプログラム  
106 カードインターフェース制御部  
110 半導体メモリカード  
111 ホストインターフェース部  
114 メモリコントローラ  
115 不揮発性メモリ  
116 メモリ  
118, 1601, 1602 ファイルシステム管理領域  
119 カード情報格納部  
120, 2901, 2902, 2903 ファイルシステムインターフェース制御部  
501 管理情報領域  
502 データ領域  
503 マスターブートレコード・パーティションテーブル  
504 パーティションブートセクタ  
505, 506 FAT  
507 ルートディレクトリエントリ  
701 ディレクトリエントリ  
1603 低レベルIOインターフェース制御部  
1701 ファイルシステム制御部  
2501 同期制御部  
2601 FAT(RAM上)  
2602, 2603, 2604, 2605 オープンファイル情報  
2606 キャッシュ管理テーブル  
2904 ファイルシステム種別フラグ

発明を実施するための最良の形態



[0013] 以下、本発明の半導体メモリ装置の実施の形態について半導体メモリカードを例として、図面を参照しつつ説明する。

(実施の形態1)

図1は本発明の実施の形態1における半導体メモリカード及びアクセス装置の要部構成図である。図1においてアクセス装置100は、CPU101、RAM102、スロット103、ROM104を含む。ROM104にはアクセス装置100を制御するプログラムが格納されている。このプログラムはアプリケーションプログラム105とカードインターフェース制御部(カードI/F制御部)106の機能を達成するための制御プログラムとを含み、RAM102を一時記憶メモリの領域として使用し、CPU101上で動作する。

[0014] スロット103は、半導体メモリカード110とアクセス装置100との接続部であり、スロット103を経由してアクセス装置100と半導体メモリカード110間で制御信号及びデータが送受信される。ROM104内のアプリケーションプログラム105は、アクセス装置100全体の制御を行い、カードインターフェース制御部106は、アクセス装置100から半導体メモリカード110に対するアクセスの制御を行う。

[0015] 一方、図1において半導体メモリカード110は、ホストインターフェース部(ホストI/F部)111、CPU112、RAM113、メモリコントローラ114、不揮発性メモリ115、メモリ116、ROM117を含む。ホストインターフェース部111は、半導体メモリカード110の外部に存在するアクセス装置100から制御信号を含むコマンドとデータとを受信すると共に、アクセス装置100に対して応答を送受するインターフェースである。ROM117には半導体メモリカード110を制御するプログラムが格納されており、このプログラムはRAM113を一時記憶領域として使用し、CPU112上で動作する。ROM117は更に、不揮発性メモリ115上のデータをファイルとして管理するためのファイルシステム制御を行うファイルシステムインターフェース制御部(ファイルシステムI/F制御部)120を含む。

[0016] メモリコントローラ114は、ホストインターフェース部111を介してアクセス装置100から制御信号を含むコマンドが入力されたとき、不揮発性メモリ115に対するデータの消去、書き込み、読み出しを制御するものである。不揮発性メモリ115は、ユーザデータ等が格納されるデータ記憶領域を有し、ファイルシステムインターフェース制御

部120により制御されるファイルシステム管理領域118を含む。メモリ116は更新可能な不揮発性メモリであり、半導体メモリカード110のカード情報を格納するカード情報格納部119を有している。カード情報格納部119は一般的には装置情報格納部である。カード情報は、後述するように不揮発性メモリ115の半導体メモリカードの物理的特性、例えば消去ブロックサイズやカード種別に関する情報である。

[0017] 続いて、半導体メモリカード110の記録素子として使用する不揮発性の半導体メモリの特徴について説明する。半導体メモリは、小型及び軽量の情報記録媒体を構成することが可能であり、様々な技術分野における情報記録媒体としての確固たる地位を築きつつある。半導体メモリはEEPROM又はフラッシュROMと呼ばれる不揮発性メモリを情報記録用の素子として使用している(以下、フラッシュメモリという)。本実施の形態の不揮発性メモリ115は、データ消去の最小単位である多数のブロックを有する。各ブロックは複数の連続する特定数のセクタからなる。不揮発性メモリ115はホストインターフェース部111を介してアクセス装置100からデータが読み書きされる。

[0018] 特に多くの不揮発性メモリで使用されるNAND型のフラッシュメモリには、データを書き込む前に一旦書き込み先に記録されているデータを消去して、未記録の状態に戻してからデータの書き込みを行わなければならないという特徴がある。ここでデータを消去する単位は消去ブロックと呼ばれ、アクセスの最小単位であるセクタが複数個、例えば連続する2の*i*乗個のセクタ(*i*は0以上の整数)が集まったブロックとして管理されている。図2はフラッシュメモリにおける消去ブロックとセクタとの関係の一例を示した説明図である。図2の例では、1つの消去ブロックは*i*=5即ち32セクタから構成されている。アクセスはセクタ(例えば512バイト)単位で行うことが可能であるが、書き込みに先立ち必要となるデータの消去処理は消去ブロック(16KB)単位で行われる。

[0019] この半導体メモリカード110におけるデータ消去及び書き込み処理の例を、図3及び図4を用いて説明する。図3及び図4は書き込み処理の一例を示すフローチャートである。特に図3は消去ブロックの倍数長のデータを書き込む場合における半導体メモリカード110内部の処理手順を示し、図4は1セクタのデータを書き込む場合における半導体メモリカード110内部の処理手順を示している。

[0020] 図3におけるデータ記録処理では、まずアクセス装置100から送信されたコマンドと引数を、ホストインターフェース部111を介して受信する(S301)。次に、受信したコマンドを参照し、自身が認識できない不正コマンドか否かを判定する(S302)。不正コマンドの場合(S302のY, Yesの意味、以下同様)、アクセス装置100にエラーを通知して処理を終了する(S303)。認識可能なコマンドの場合(S302のN, Noの意味、以下同様)、そのコマンドが書き込みコマンドであるかを判定する(S304)。書き込みコマンド以外の場合、各コマンドに対応した他の処理を実施する(S305)。書き込みコマンドの場合、コマンドと共に指定された引数が正しいかを判定する(S306)。不正な引数と判定した場合、アクセス装置100にエラーを通知して処理を終了する(S307)。

[0021] 引数が正しいと判定した場合、引数に格納された書き込み位置や書き込みサイズの情報から、実際にフラッシュメモリにデータを書き込む消去ブロックの物理アドレスを決定する(S308)。次いで、書き込みに先立ち、メモリコントローラ114を介して、フラッシュメモリに存在し、S308で決定した消去ブロック(物理ブロック)に存在するデータを消去する(S309)。次に、アクセス装置100から1セクタ分のデータを、ホストインターフェース部111を介して受信する(S310)。データの受信を完了すると、受信した1セクタ分のデータを、メモリコントローラ114を介して不揮発性メモリ115へ書き込む(S311)。S310のデータ受信処理及びS311の書き込み処理を、1消去ブロック分のデータ書き込みが完了するまで繰り返し実施する(S312)。S308からS312までの1消去ブロック分のデータ書き込み処理を、アクセス装置100から指定された書き込みサイズ分のデータ書き込みが完了するまで繰り返し実施する(S313)。アクセス装置100から指定された書き込みサイズ分のデータ書き込みが完了した場合は処理を終了する。

[0022] 図4におけるデータ記録処理において、S401〜S411の処理は図3のS301〜S311の処理と同じである。図3の処理と異なる点は、S412において書き込みを行う消去ブロックに含まれるデータのうち、アクセス装置100からデータを受信する1セクタ以外のデータを、S408で決定した消去ブロックに書き込む点である。NAND型のフラッシュメモリでは、データ書き込みに先立ち一旦データを消去する必要がある。この

消去処理は消去ブロック単位で行えないため、1セクタのデータだけを書き込む場合でも、1消去ブロック分のデータ消去が必要である。更にS412の処理に示すように、データが更新されたセクタと同一の消去ブロックに含まれる他のセクタの既存データを新しい消去ブロックに書き戻す必要がある。

- [0023] 図3及び図4で示したようにデータ記録処理では、大きく分けてコマンド解釈処理、データ消去処理、データ書き込み処理の3つの処理が存在する。例えば、コマンド解釈のオーバーヘッドに3m秒、1セクタの書き込み処理に200 $\mu$ 秒、1消去ブロック(16KB)の消去処理に2m秒かかるフラッシュメモリを想定する。このフラッシュメモリの1消去ブロック(16KB)の書き込みでは図3に示す処理が実行され、コマンド解釈に3m秒、消去処理に2m秒、書き込み処理に32 $\times$ 200 $\mu$ 秒かかり、合計11.4m秒となる。
- [0024] 同様に1セクタ(512B)の書き込みでは図4に示す処理が実行され、コマンド解釈に3m秒、消去処理に2m秒、書き込み処理に200 $\mu$ 秒+31 $\times$ 200 $\mu$ 秒かかり、合計11.4m秒となる。すなわち、16KBのデータを書き込んだ場合と1セクタのデータを書き込んだ場合では、ほぼ同じ時間がかかることになる。この例ではデータ転送時間等を考慮せず極端に性能差が出る場合について説明したが、実際のフラッシュメモリにおいても消去ブロック単位で書き込みを行った場合に書き込み時間が最短になる。
- [0025] また、半導体メモリカード110の記録素子として使用するフラッシュメモリの枚数は1枚とは限らず、複数枚使用して並列処理することにより、アクセス性能を高めた半導体メモリカード110が存在する。このような半導体メモリカード110では、複数の消去ブロックを管理単位としてフラッシュメモリを制御しており、この管理単位で書き込みを行った場合に書き込み時間が最短となる。このように半導体メモリカード110のアクセス特性は、使用するフラッシュメモリの種類や枚数、フラッシュメモリの管理方法等に依存し、半導体メモリカード110の世代や製造者の違いにより、アクセス特性は異なる。カード情報格納部119に格納されるカード情報は、物理的特性とカード種別情報を有する。物理的特性はコマンド解釈のオーバーヘッド、1セクタの書き込み処理時間、1消去ブロックの消去時間、消去ブロックサイズ等や最適アクセス単位の情報、ある

いは最適アクセス単位を決定するために必要なカード情報を含み、カード種別情報は半導体メモ리카ードのバージョン情報を含む。

- [0026] 続いて、半導体メモ리카ード110内の不揮発性メモリ115に格納されたデータを管理するファイルシステムの一例として、FATファイルシステムを説明する。図5にFATファイルシステムの構成を示す。ファイルシステム管理領域118は、半導体メモ리카ード110内の不揮発性メモリ115におけるファイルシステムで管理する領域を意味する。FATファイルシステムでは、ファイルシステム管理領域118の先頭にファイルシステム管理領域の全体を管理するための管理情報領域501が存在し、引き続いてファイル内のユーザデータ等を格納するデータ領域502が存在する。管理情報領域501は、マスターブートレコード・パーティションテーブル(MBR・PT)503、パーティションブートセクタ(PBS)504、FAT505、FAT506、ルートディレクトリエントリ(RDE)507から構成される。
- [0027] マスターブートレコード・パーティションテーブル503は、ファイルシステム管理領域を複数のパーティションと呼ばれる領域に分割して管理するための情報を格納する部分である。パーティションブートセクタ504は、1つのパーティション内の管理情報を格納する部分である。FAT505、506は、ファイルに含まれるデータの物理的な格納位置を示す部分である。ルートディレクトリエントリ507は、ルートディレクトリ直下に存在するファイルやディレクトリの情報を格納する部分である。また、FAT505、506は、ファイルに含まれるデータの物理的な格納位置を示す重要な領域であることから、通常はファイルシステム管理領域118内に2つの同じ情報を持つFAT505、506を設けることにより二重化されている。
- [0028] データ領域502は複数のクラスタに分割され管理されており、各クラスタにはファイルに含まれるデータが格納されている。多くのデータを格納するファイル等は、複数のクラスタに跨ってデータを格納しており、各クラスタ間の繋がりは、FAT505、506に格納されたリンク情報により管理されている。
- [0029] 次に図6、図7、図8を用いてFATファイルシステムにおけるファイルデータの書き込み例を説明する。図6はFATファイルシステムにおけるファイルデータの書き込み処理手順を示したフローチャートである。図7、図8はそれぞれ書き込み処理前、書き

込み処理後のディレクトリエントリ701、FAT505、506、データ領域502の一例を示した説明図である。FATファイルシステムでは、ルートディレクトリエントリ507やデータ領域502の一部に、ファイル名やファイルサイズ、ファイル属性等の情報を格納したディレクトリエントリ701が格納されており、図7(a)にディレクトリエントリ701の一例が示されている。このディレクトリエントリ701で示されるファイル例では、ファイル名がFILE1. TXTであり、クラスタ番号10からのファイルのデータが格納されている。また、ファイルサイズは16000バイトである。図7では1クラスタの大きさを4096バイトと仮定しており、4クラスタに跨ってファイルのデータが格納されている。

- [0030] 図6を用いてファイルデータの書き込み処理を説明する。ファイルデータ書き込み処理では、まず始めに対象ファイルのディレクトリエントリ701を読み込む(S601)。次に、読み込んだディレクトリエントリ701に格納されたファイルの開始クラスタ番号を取得し、ファイルデータの先頭位置を確認する(S602)。次に、FAT505、506を読み出し、S602で取得したファイルデータの先頭位置から順にFAT505、506上でリンクを辿り、書き込み位置のクラスタ番号を取得する(S603)。
- [0031] 次に、データ書き込みに際し、ファイルに新たに空き領域を割り当てる必要があるか判定する(S604)。空き領域の割り当てが不要な場合S606の処理に進む。空き領域の割り当てが必要な場合、FAT505、506上で空き領域を検索し、1クラスタの空き領域をファイルの終端に割り当てる(S605)。次に、現在参照しているクラスタ内に書き込めるだけのデータをデータ領域502に書き込む(S606)。次に、全データの書き込みが完了したかを判定する(S607)。まだデータが残っている場合、S604の処理に戻る。全データの書き込みが完了した場合、ディレクトリエントリ701内に格納されたファイルサイズやタイムスタンプ等を更新し、半導体メモ리카ード110に書き込む(S608)。最後にFAT505、506を半導体メモ리카ード110に書き込み、処理を完了する(S609)。
- [0032] このようなファイルデータの書き込み処理により、図7に示された16000バイトのデータを持つFILE1. TXTに1000バイトのデータを更に書き込んだ場合、図8に示されるような17000バイトのデータを持つファイルに変化する。
- [0033] このようにFATファイルシステムでは、ファイルデータの格納領域としてクラスタ単位

で領域割り当てが行われ、データが格納される。また、1つのファイルに割り当てられる複数のクラスタは必ずしも連続しているとは限らず、不連続な領域が割り当てられる可能性がある。最悪の場合、クラスタ単位に分割された不連続領域にファイルデータが書き込まれることになる。この場合、半導体メモ리카ード110に対する1回のアクセスサイズは1クラスタ以下の大きくなり、半導体メモ리카ード110に対して最も高速にアクセスするために必要なアクセス単位がクラスタサイズよりも大きい場合、半導体メモ리카ード110の最高性能でアクセスすることができなくなる。このような問題はFATファイルシステムに限らず、他のファイルシステムにおいても発生する。

- [0034] そのため本実施の形態では、半導体メモ리카ード110の物理的特性を含むカード情報を格納するカード情報格納部119を設ける。そしてそのカード情報を基に半導体メモ리카ード110の物理的特性に適したファイルアクセスを行うために、ファイルシステムインターフェース制御部120を半導体メモ리카ード110内に備える。このことにより、アクセス装置100が半導体メモ리카ード110のアクセス特性を意識することなく、最適なファイルアクセスを行うことができる。
- [0035] 続いて本実施の形態におけるファイルシステムインターフェース制御部120の機能について説明する。本実施の形態では、半導体メモ리카ード110内の不揮発性メモリ115上に構築されたファイルシステムを、半導体メモ리카ード110内のファイルシステムインターフェース制御部120が制御する。そのため、アクセス装置100は、半導体メモ리카ード110内の不揮発性メモリ115に格納されたファイルにアクセスするために、ファイルアクセス要求を半導体メモ리카ード110に発行する。
- [0036] 図9は、ファイルシステムインターフェース制御部120がアクセス装置100から受け付ける上位のコマンド群(以下、ファイルシステムAPIという)の一覧を示した図である。図9に示すように、ファイルシステムインターフェース制御部120は、アクセス装置100から、OPEN(ファイルのオープン)や、CLOSE(ファイルのクローズ)、READ(ファイルからデータを読み込む)、WRITE(ファイルへデータを書き込む)等のファイルアクセス要求を受け付け、アクセス装置100に対してファイルシステムの機能を提供する。即ちファイルシステムインターフェース制御部120は、カード情報格納部119に格納されたカード情報に基づいて、不揮発性メモリ115内に格納したデータをフ

イルとして管理し、ホストインターフェース部111を介してアクセス装置100から入力されたコマンドに基づいて、不揮発性メモリ115上のファイルに対するオープン、クローズ、リード、ライトを含むファイルアクセス処理を実施する。

[0037] このような方法により、アクセス装置100側にはファイルシステムを備える必要がなくなる。ファイルシステムの機能は全て半導体メモリカード110側から提供されることになる。

[0038] 次に図10～図14を用いて、アクセス装置100がファイルシステムインターフェース制御部120を介してファイルにアクセスする例を説明する。ここでは、ファイルにアクセスする例として、ルートディレクトリ直下にファイルを作成し、データを書き込む例について説明する。図10はアクセス装置100側の処理手順を示すフローチャートである。図11、図12、図13、図14は夫々アクセス装置100が半導体メモリカード110に発行するカード種別取得コマンド、OPENコマンド、WRITEコマンド、CLOSEコマンドにおける半導体メモリカード110側の処理手順を示したフローチャートである。

[0039] まず始めに図10を用いてアクセス装置100側の処理手順を説明する。まずアクセス装置100は半導体メモリカード110に対し、半導体メモリカード110のバージョン等の情報であるカード種別情報を取得するために、カード種別取得コマンドを発行する(S1001)。次に、発行したコマンドにより半導体メモリカード110からカード種別情報が取得できたかを判定する(S1002)。取得に失敗した場合はエラーが発生したと判断し、処理を終了する(S1003)。

[0040] 取得に成功した場合、半導体メモリカード110がファイルシステムAPIに対応しているかを判定する(S1004)。対応していない半導体メモリカード110であれば、ファイルシステムAPIを用いたアクセスが行えないため、処理を終了する(S1005)。対応している半導体メモリカード110であれば、ファイルを作成するために、半導体メモリカード110に対し、ファイル名を指定してOPENコマンドを発行する(S1006)。

[0041] 次に、発行したOPENコマンドによる処理が成功したかを判定する(S1007)。処理に失敗した場合はエラー終了する(S1008)。処理に成功した場合、OPENコマンドのレスポンスとして、作成したファイルにアクセスするために使用するファイルハンドル(識別子)を半導体メモリカード110から取得する。次にファイルに格納するデータを



作成する(S1009)。次に、作成したデータをファイルに書き込むために、OPENコマンドにより取得したファイルハンドル、書き込みサイズ、作成したデータ等を指定し、半導体メモリカード110に対しWRITEコマンドを発行する(S1010)。

[0042] 次に、発行したWRITEコマンドによる処理が成功したかを判定する(S1011)。処理に失敗した場合はエラー終了する(S1012)。処理に成功した場合、ファイルハンドルを指定し、半導体メモリカード110に対しCLOSEコマンドを発行する(S1013)。次に、発行したCLOSEコマンドによる処理が成功したかを判定する(S1014)。処理に失敗した場合はエラー終了する(S1015)。処理に成功した場合はファイル作成及びデータ書き込み処理を終了する。

[0043] 上記処理手順の内、S1001、S1006、S1010、S1013のコマンド発行処理は、アクセス装置100内のカードインターフェース制御部106が実施し、その他の処理はアプリケーションプログラム105が実施する。

[0044] 次に、図11から図14を用いて半導体メモリカード110側の処理手順を説明する。図11はカード種別取得コマンド発行時の半導体メモリカード110側の処理手順を示したフローチャートである。まず半導体メモリカード110はアクセス装置100からコマンドを受信する(S1101)。次に、受信したコマンドを参照し、自身が認識できない不正コマンドか否かを判定する(S1102)。不正コマンドの場合、アクセス装置100にエラーを通知して処理を終了する(S1103)。認識可能なコマンドの場合、そのコマンドがカード種別取得コマンドであるかを判定する(S1104)。カード種別取得コマンド以外の場合、各コマンドに対応した他の処理を実施する(S1105)。カード種別取得コマンドの場合、カード情報格納部119に格納されたカード種別情報を読み出す(S1106)。最後に読み出したカード種別情報をアクセス装置100に送信し、ここでの処理を終了する(S1107)。

[0045] 図12はOPENコマンド発行時の半導体メモリカード110側の処理手順を示したフローチャートである。まず半導体メモリカード110はアクセス装置100からコマンドを受信する(S1201)。次に、受信したコマンドを参照し、自身が認識できない不正コマンドか否かを判定する(S1202)。不正コマンドの場合、アクセス装置100にエラーを通知して処理を終了する(S1203)。認識可能なコマンドの場合、そのコマンドがOP

ENコマンドであるかを判定する(S1204)。

- [0046] OPENコマンド以外の場合、各コマンドに対応した他の処理を実施する(S1205)。OPENコマンドの場合、コマンドと共に渡された引数が正しいか判定する(S1206)。引数で不正なファイル名を指定した場合等、OPEN処理を実施できないと判断した場合、アクセス装置100にエラーを通知して処理を終了する(S1207)。引数が正しい場合、ルートディレクトリエントリ領域(RDE)をRAM113に読み込む(S1208)。
- [0047] 次に、RAM113に読み込んだルートディレクトリエントリ領域内で空きディレクトリエントリを取得する(S1209)。取得に失敗した場合、アクセス装置100にエラーを通知して処理を終了する(S1211)。取得に成功した場合、OPENコマンドの引数で指定されたファイル名を持つファイルのディレクトリエントリ(DE)をRAM113上に作成する(S1212)。次に、RAM113上に作成したディレクトリエントリを不揮発性メモリ115に書き込む(S1213)。
- [0048] 次に、作成したファイルにアクセスするために必要な情報をオープンファイル情報(OFI)として、RAM113上に作成する(S1214)。このオープンファイル情報には、ファイル名、ファイルサイズ、ディレクトリエントリの位置、読み出し専用オープン・書き込み専用オープン等のオープン属性、ファイル内の現在参照位置等が含まれる。最後に、オープンファイル情報を一意に識別するファイルハンドルを作成し、アクセス装置100に通知して処理を終了する(S1215)。OPEN処理後のファイルアクセスは、このファイルハンドルを使用してアクセスする。
- [0049] 図13はWRITEコマンド発行時の半導体メモリカード110側の処理手順を示した図である。まず半導体メモリカード110はアクセス装置100からコマンドを受信する(S1301)。次に、受信したコマンドを参照し、自身が認識できない不正コマンドか否かを判定する(S1302)。不正コマンドの場合、アクセス装置100にエラーを通知して処理を終了する(S1303)。認識可能なコマンドの場合、そのコマンドがWRITEコマンドであるかを判定する(S1304)。WRITEコマンド以外の場合、各コマンドに対応した他の処理を実施する(S1305)。
- [0050] WRITEコマンドの場合、コマンドと共に渡された引数が正しいか判定する(S1306)。引数で不正なファイルハンドルを指定した場合等、WRITE処理を実施できないと

判断した場合、アクセス装置100にエラーを通知して処理を終了する(S1307)。引数が正しい場合、ファイルハンドルを基にRAM113上のオープンファイル情報(OFI)を特定する(S1308)。次に、特定したオープンファイル情報を参照し、ファイル内の現在参照位置を確認する(S1309)。現在参照位置は次にデータを書き込む位置であり、ファイルの終わりから追記する場合には空き領域の取得が必要となる。

[0051] 次に、データ書き込みに先立ち、空き領域の取得が必要かを判定する(S1310)。空き領域の取得が不要な場合、S1315の処理に進む。空き領域の取得が必要な場合、RAM113上のFATを参照し、空きクラスタを取得する(S1311)。ここでは、FATが予め不揮発性メモリ115からRAM113に読み込まれていると想定する。取得に失敗した場合、アクセス装置100にエラーを通知して処理を終了する(S1313)。取得に成功した場合、RAM113上でFATを更新し、S1311で取得した空きクラスタを使用中クラスタに変更する(S1314)。

[0052] 次に、不揮発性メモリ115内のS1311で取得した空きクラスタ位置に、データを書き込む(S1315)。S1310からS1315までの処理を、全データの書き込みが完了するまで繰り返す(S1316)。全データの書き込みが完了した場合、RAM113上のオープンファイル情報に格納されたファイルサイズ等の情報を更新する(S1317)。最後に処理完了をアクセス装置100に通知して処理を終了する(S1318)。

[0053] 図14はCLOSEコマンド発行時の半導体メモリカード110側の処理手順を示したフローチャートである。まず半導体メモリカード110はアクセス装置100からコマンドを受信する(S1401)。次に、受信したコマンドを参照し、自身が認識できない不正コマンドか否かを判定する(S1402)。不正コマンドの場合、アクセス装置100にエラーを通知して処理を終了する(S1403)。認識可能なコマンドの場合、そのコマンドがCLOSEコマンドであるかを判定する(S1404)。

[0054] CLOSEコマンド以外の場合、各コマンドに対応した他の処理を実施する(S1405)。CLOSEコマンドの場合、コマンドと共に渡された引数が正しいかを判定する(S1406)。引数で不正なファイルハンドルを指定した場合等、CLOSE処理を実施できないと判断した場合、アクセス装置100にエラーを通知して処理を終了する(S1407)。引数が正しい場合、ファイルハンドルを基にRAM113上のオープンファイル情報を

特定する(S1408)。次に、特定したオープンファイル情報を参照し、オープンファイルのディレクトリエントリの格納位置を特定し、ディレクトリエントリを含む1セクタ分のデータをRAM113に読み込む(S1409)。

[0055] 次に、オープンファイルのディレクトリエントリに格納されたファイルサイズやタイムスタンプ等をRAM113上で更新する(S1410)。次に、RAM113上のディレクトリエントリを不揮発性メモリ115に書き込む(S1411)。次に、RAM113上のFATを不揮発性メモリ115に書き込む(S1412)。ディレクトリエントリ、FATの書き込みを完了した後、RAM113上のオープンファイル情報をクリアし、ファイルがオープンされていない状態に変更する(S1413)。最後に処理完了をアクセス装置100に通知して処理を終了する(S1414)。

[0056] このような本実施の形態によれば、アクセス装置100はファイルシステムAPIのコマンドを発行するのみであり、ファイルシステムの制御は半導体メモリカード110側で実施される。そのため、アクセス装置100が半導体メモリカード110の特性を認識する必要はなく、半導体メモリカード110内で最適なファイルアクセスを実施することができる。

[0057] 続いて、半導体メモリカード110のファイルシステムインターフェース制御部120が、半導体メモリカード110の物理的特性に応じたファイルアクセスを行う点について更に説明する。本実施の形態では、ファイルシステムインターフェース制御部120がカード情報格納部119から半導体メモリカード110の物理的特性に関するカード情報を取得し、その情報を基にファイルアクセスを行う。これにより、半導体メモリカード110の物理的特性に応じた最適なファイルアクセスを実現する。以下に、カード情報格納部119から取得したカード情報を基にファイルアクセスを行う方法として、2つの例を説明する。

[0058] 第1の例として、半導体メモリカード110の最適アクセス単位を意識したファイルデータ書き込み方法について説明する。半導体メモリカード110に高速にアクセスするための最適なアクセス単位は、使用するフラッシュメモリの種類や枚数、フラッシュメモリの管理方法等に依存し、半導体メモリカード110の世代や製造者の違いにより異なる。本実施の形態では、カード情報格納部119に最適アクセス単位の情報、あるいは

は最適アクセス単位を決定するために必要なカード情報を格納し、ファイルシステムインターフェース制御部120が最適アクセス単位を意識したファイルアクセスを行う。最適アクセス単位とは不揮発性メモリ115に最も効率よくアクセスできる単位である。1枚の不揮発性メモリチップを用いる場合には、消去ブロックがそのまま最適アクセス単位である。又複数、例えば8枚の不揮発性メモリチップを並列に使用する場合には、各メモリチップの消去ブロックの8倍、例えば $16\text{KB} \times 8 = 128\text{KB}$ が最適アクセス単位となる。

[0059] 図15は、不揮発性メモリ115に構築されたファイルシステムにおけるデータ領域502のデータ配置の一例を示した図である。この例では、ファイルシステムが1クラスタを16KBとして管理しており、最適アクセス単位が128KBである半導体メモリカード110を想定している。従来のファイルシステムでは、ファイルデータを書き込む際の空き領域取得において最適アクセス単位を意識せず、任意の空き領域をファイルに割り当てるため、例えば図15のクラスタ番号4の領域を割り当て、クラスタ単位でデータを書き込む。これに対し、本実施の形態におけるファイルアクセスでは、最適アクセス単位を意識し、最適アクセス単位の空き領域割り当てを行う。

[0060] そのため、図15の最適アクセス単位0や最適アクセス単位1のように、最適アクセス単位内の一部が使用されている領域はファイルデータに割り当てず、最適アクセス単位2のように、最適アクセス単位内の全てが空きである領域を割り当て、最適アクセス単位でデータを書き込む。この最適アクセス単位の空き領域割り当ては、図13で説明したWRITEコマンド処理手順におけるS1311の処理で具体的に実施される。このような領域割り当てを行うことにより、ある程度大きなファイルサイズを有するファイルの場合、アクセス装置側では何らの対応をすることなく最適アクセス単位で半導体メモリカード110にアクセスすることができる。従って半導体メモリカード110に高速にファイルアクセスすることが可能となる。尚この場合にはデータの書き込み毎に最適アクセス単位で新しい領域が用いられるため、処理が行われていないバックグラウンドで空き領域を含む最適アクセス単位に散在するデータをまとめるデフラグ処理を行うことが考えられる。

[0061] 第2の例として、半導体メモリカード110の最適アクセス単位を意識したディレクトリ

領域の割り当て方法について説明する。ディレクトリを新規に作成した場合や、ファイルを新規に作成し、ディレクトリ領域に新たな領域を割り当てる必要が生じた場合、従来のファイルシステムでは、ディレクトリ領域を割り当てる際の空き領域取得において、先の例と同様任意の空き領域を割り当てるため、例えば図15のクラスタ番号18の領域を割り当てる。この場合、クラスタ番号18の領域をディレクトリ領域として使用することにより、最適アクセス単位2内に使用領域が含まれることになり、第1の例で説明したファイルデータ領域の割り当てに使用できなくなる。

[0062] そのため、本実施の形態におけるファイルシステムインターフェース制御部120は、最適アクセス単位内の一部にディレクトリ領域が既に割り当てられている領域を優先的にディレクトリ領域の割り当て領域として使用し、可能な限り同一最適アクセス単位内にディレクトリ領域が含まれるようにする。すなわち、図15の例では、既にディレクトリ領域が割り当てられている最適アクセス単位1に含まれるクラスタ番号12から17の領域をディレクトリ領域として割り当てる。これにより、最適アクセス単位の連続空き領域が生成され易くなり、第1の例のファイルデータ割り当てが効果的に行えるようになる。これにより高速にファイルアクセスすることができる。

[0063] 以上説明したように、本実施の形態では、半導体メモリカード110の物理的特性を含むカード情報を格納するカード情報格納部119と、カード情報を基に半導体メモリカード110の物理的特性に適したファイルアクセスを行うファイルシステムインターフェース制御部120とを半導体メモリカード110内に備える。これにより、アクセス装置100が半導体メモリカード110のアクセス特性を意識することなく、最適なファイルアクセスを行うことが可能となる。これによりアクセス装置側は最適アクセスするために種々の半導体メモリカードに対応する必要がなくなり、対応のための検証作業の量も減少する。

[0064] また、本実施の形態で説明したファイルシステムAPIを解釈可能なアクセス装置100は、半導体メモリカード110上に構築されているファイルシステムの種類を認識する必要はなく、異なるファイルシステムにより管理されている複数の半導体メモリカード110にアクセスすることが可能となる。即ち半導体メモリカード110上に構築されているファイルシステムの種別に関わらず、データを授受することが可能となる。

- [0065] 尚、図10から図14で説明したアクセス装置100及び半導体メモリカード110の処理手順は一例であり、アクセス装置100側からファイルシステムAPIコマンドを発行し、半導体メモリカード110側でファイルシステム制御を行う方法であれば、上記の実施の形態と異なる処理手順を用いても良い。
- [0066] また本実施の形態では、カード情報格納部119に格納されたカード情報を基に、ファイルシステムインターフェース制御部120が最適アクセス単位を決定する例について説明した。しかし、カード情報格納部119に格納されたカード情報を基にファイルシステムインターフェース制御部120が最適アクセス単位を取得できる方法であれば、他の方法を用いても構わない。例えばカード情報格納部119に格納されたカード情報をアクセス装置100が読み出し、アクセス装置100側で最適アクセス単位を決定してファイルシステムインターフェース制御部120に通知する等が考えられる。
- [0067] また、図9で示したファイルシステムAPIは一例であり、図9の一部のAPIのみを選択して使用しても良いし、その他のファイルシステム制御に関するAPIを加えて使用しても良い。また、本実施の形態ではFATファイルシステムを例として説明したが、その他のファイルシステムを使用しても構わない。また、カード情報格納部119に格納されたカード情報は、半導体メモリカード120の使用状態に応じて値を更新可能としても良い。また、カード情報格納部119を含むメモリ116は不揮発性メモリ115に含めても良いし、カード情報を更新しない場合はROM117に含めても良い。
- [0068] 以上のように実施の形態1は、半導体メモリカード内にユーザデータ等を格納する不揮発性メモリと、半導体メモリカードのカード情報を格納するカード情報格納部と、カード情報格納部に格納されたカード情報を基に、半導体メモリカードの特性に適したファイルアクセスを行うファイルシステムインターフェース制御部とを備えることを特徴とする。
- [0069] (実施の形態2)
- 次に本発明の実施の形態2における半導体メモリカードについて説明する。図16は本実施の形態(その1)における半導体メモリカード及びアクセス装置の構成図である。また図17は本実施の形態(その2)における半導体メモリカード及びアクセス装置の構成図である。図16に示す半導体メモリカード110と図1に示す半導体メモリカ

ード110と異なる点は、本実施の形態では不揮発性メモリ115内の領域が独立した2つの論理アドレス群に分割されており、ファイルシステム管理領域A(第1の領域)1601と、ファイルシステム管理領域B(第2の領域)1602とが存在する点である。更に半導体メモリカード110のROM117内に、ファイルシステムインターフェース制御部120に加えて、低レベルIOインターフェース制御部1603を設けたことである。

[0070] また、図17に示す構成において図16と異なる点は、アクセス装置100のROM104内に、アプリケーションプログラム105とカードインターフェース制御部106に加えて、ファイルシステム制御部1701が存在する点である。このアクセス装置100は、従来の半導体メモリカード(ファイルシステムインターフェース制御部が設けられていないもの)にデータを記録したり、データを読み出す場合に必要となる構成である。

[0071] 図16に示すファイルシステムインターフェース制御部120は、カード情報格納部119に格納されたカード情報に基づいて不揮発性メモリ115内に格納したデータをファイルとして管理し、ホストインターフェース部111を介してアクセス装置100から不揮発性メモリ115上のファイルに対するオープン、クローズ、リード、ライトを含むファイルアクセス処理を要求するコマンドが入力されたとき、不揮発性メモリ115内に存在するファイルに対するファイルアクセス処理を実施するものである。

[0072] 低レベルIOインターフェース制御部1603は、ホストインターフェース部111を介してアクセス装置100から不揮発性メモリ115の第2の領域1602に対するデータ書き込み、又は読み出し処理を要求するコマンドが入力されたとき、不揮発性メモリ115内の第2の領域1602に対するデータ書き込み、又は読み出し処理を実施するものである。ファイルシステムインターフェース制御部120と異なる点は、低レベルIOインターフェース制御部1603が不揮発性メモリ115の第2の領域1602に対するデータ読み出し及び書き込み等のアクセス制御のみを行い、ファイルシステムの制御は行わない点である。

[0073] 低レベルIOインターフェース制御部1603を介して半導体メモリカード110にアクセスする場合に、アクセス装置100の構成は図17のようになる。この場合は、図17に示すようにアクセス装置100側のファイルシステム制御部1701が半導体メモリカード110におけるファイルシステムの制御を行う。一方、ファイルシステムインターフェース制



御部120を介して半導体メモリカード110にアクセスするアクセス装置100は、図16で示した構成をとる。この場合は本発明の実施の形態1と同様に、アクセス装置100側にファイルシステム制御部1701を備えないものとする。

[0074] また図16に示すように、不揮発性メモリ115の領域はファイルシステム管理領域A1601と、ファイルシステム管理領域B1602の2つの領域に分割されており、ファイルシステム管理領域A1601はファイルシステムインターフェース制御部120により制御され、ファイルシステム管理領域B1602は低レベルIOインターフェース制御部1603により制御される。

[0075] 即ち本実施の形態では、半導体メモリカード110はアクセス装置100からのアクセスを受け付けるインターフェース制御部を2種類備えており、夫々のインターフェース制御部経由でアクセスされる領域は独立している。これにより、2つのインターフェース制御部のいずれか一方のみを解釈可能なアクセス装置100からも、半導体メモリカード110にアクセスすることが可能となり、アクセス装置の互換性を高めることができる。

[0076] 続いて、本実施の形態における低レベルIOインターフェース制御部1603の機能について説明する。本実施の形態では、半導体メモリカード110内の不揮発性メモリ115を2つの管理領域A、Bに分割し、ファイルシステム管理領域A1601をファイルシステムインターフェース制御部120により制御し、ファイルシステム管理領域B1602を低レベルIOインターフェース制御部1603により制御する。ファイルシステムインターフェース制御部120を介してアクセスする方法は、実施の形態1で説明した方法と同様であるので説明を割愛する。ここでは低レベルIOインターフェース制御部1603を介してアクセスする方法について説明する。

[0077] 図18は低レベルIOインターフェース制御部1603がアクセス装置100から受け付けるコマンド群(以下、低レベルIOAPIという)の一覧を示した図である。図18に示すように、低レベルIOインターフェース制御部1603は、アクセス装置100から、RAW\_READ(データを読み込む)や、RAW\_WRITE(データを書き込む)等の図9に示すファイルシステムよりも低レベルな入出力要求を受け付け、アクセス装置100に対し、これら低レベルIOの機能を提供する。即ち本実施の形態において低レベルIO

インターフェース制御部1603を介してアクセスするアクセス装置100には、前述したようにファイルシステムを制御するためのファイルシステム制御部1701を備える必要がある。

[0078] 次に図19～図22を用いて、アクセス装置100が低レベルIOインターフェース制御部1603を介してファイルにアクセスする例を説明する。ここでは、ファイルにアクセスする例として、ルートディレクトリ直下にファイルを作成し、データを書き込む例について説明する。図19はアクセス装置100側の全体の処理手順を示すフローチャートである。図20、図21、図22は夫々アクセス装置100内のファイルシステム制御部1701が行うファイルオープン処理、データ書き込み処理、ファイルクローズ処理の処理手順を示したフローチャートである。

[0079] まず始めに図19を用いてアクセス装置100側の全体の処理手順を説明する。図19の処理手順が図10の処理手順と異なる点は、ファイルオープン、データ書き込み、ファイルクローズの各処理を半導体メモリカード110側で処理させるのではなく、アクセス装置100内のファイルシステム制御部1701で処理する点である。具体的には、図19のS1906、S1910、S1913の各処理は、後述の図20～図22に示す処理手順で、アクセス装置100内のファイルシステム制御部1701が処理する。即ち図19において、S1901はカードインターフェース制御部106が実施し、S1906、S1910、S1913の各処理はファイルシステム制御部1701が実施し、その他の処理はアプリケーションプログラム105が実施する。

[0080] 次に、図20～図22を用いてアクセス装置100内のファイルシステム制御部1701が行う処理手順を説明する。カード種別取得コマンド発行時の半導体メモリカード110側の処理手順は図11と同様であり、ここでは説明を割愛する。図20～図22に示すファイルオープン処理、データ書き込み処理、ファイルクローズ処理の処理手順は、図12～図14で説明した処理手順とほぼ同じであり、ここでは相違点のみを説明する。

[0081] 第1の相違点は、コマンド受信からコマンド解釈までの処理(図12におけるS1201からS1205までの処理)が存在しない点である。これらの処理は半導体メモリカード110側でコマンドを解釈するための処理であり、アクセス装置100内のファイルシステ

ム制御部1701で実施する処理の場合不要であるため、処理手順から削除されている。

[0082] 第2の相違点は、不揮発性メモリ115に対しデータを読み書きする処理を、RAW\_READコマンド発行、又はRAW\_WRITEコマンド発行と記載し、各コマンドの成否判定を追加している。実施の形態1では、半導体メモリカード110のファイルシステムインターフェース制御部120が不揮発性メモリ115にアクセスしていたのに対し、本実施の形態では、アクセス装置100のカードインターフェース制御部106が低レベルIOAPIコマンドを発行し、半導体メモリカード110の低レベルIOインターフェース制御部1603を介してアクセスする。そのため、不揮発性メモリ115にアクセスする処理が低レベルIOAPIコマンド発行処理に変更されている。

[0083] 以上説明したように、本実施の形態では、半導体メモリカード110の物理的特性を含むカード情報を格納するカード情報格納部119と、その情報を基に半導体メモリカード110の物理的特性に適したファイルアクセスを行うファイルシステムインターフェース制御部120と、低レベルIO要求を受け付ける低レベルIOインターフェース制御部1603と、不揮発性メモリ115とを、半導体メモリカード110内に備える。更に不揮発性メモリ115内の領域を2つに分割し、夫々の領域をファイルシステムインターフェース制御部120、低レベルIOインターフェース制御部1603からアクセスする。このような構成をとることで、2つのインターフェースのいずれか一方のみを解釈可能なアクセス装置、例えば従来のアクセス装置からでも半導体メモリカード110にアクセスすることが可能となる。

[0084] 尚、本実施の形態では、低レベルIOインターフェース制御部1603を介したファイルアクセスについて図20～図22を用いて説明したが、これらの処理は従来のアクセス装置100において、ファイルシステムをアクセス装置100側で制御する場合の処理と同様のものである。本実施の形態の主眼とする点は、半導体メモリカード110側に2つのインターフェース制御部120、1603を設け、アクセス装置100からいずれのインターフェース制御部を介しても、半導体メモリカード110にアクセスできるようにした点にある。そのため、図20～図22の処理手順は一例であり、アクセス装置100側にファイルシステム制御部1701を設け、低レベルIOインターフェース制御部1603を介し

て半導体メモリカード110にアクセスする制御方法であれば、異なる処理手順を用いても良い。

[0085] また、図18で示した低レベルIOAPIは一例であり、図18の一部のAPIのみを選択して使用しても良いし、その他の低レベルIOに関するAPIを加えて使用しても良い。また、本実施の形態ではFATファイルシステムを例として説明したが、その他のファイルシステムを使用しても構わない。また、カード情報格納部119に格納されたカード情報は、半導体メモリカード120の使用状態に応じて値を更新可能としても良い。また、カード情報格納部119を含むメモリ116は、不揮発性メモリ115に含めても良いし、カード情報を更新しない場合はROM117に含めても良い。

[0086] 以上のように実施の形態2は、半導体メモリカード内に領域を2つに分割したユーザデータ等を格納する不揮発性メモリと、半導体メモリカードの物理的特性に関するカード情報を格納するカード情報格納部と、カード情報格納部に格納された情報を基に半導体メモリカードの物理的特性に適したファイルアクセスを行うファイルシステムインターフェース制御部と、半導体メモリカード外部のアクセス装置から半導体メモリカード内の不揮発性メモリに対する低レベル入出力要求を処理する低レベルIOインターフェース制御部とを備え、これらのインターフェース制御部により個々の領域を管理することを特徴とする。

[0087] (実施の形態3)

次に本発明の実施の形態3における半導体メモリカードについて説明する。図23は実施の形態3における半導体メモリカード110及びアクセス装置100の構成図である。図23に示すアクセス装置100は図17に示すものと構成が同じである。半導体メモリカード110の構成で図16に示すものと異なる点は、不揮発性メモリ115内の領域が2つに分割されておらず、単一のファイルシステム管理領域118から構成されている点である。

[0088] 即ち本実施の形態3では、実施の形態2と同様に、半導体メモリカード110内にファイルシステムインターフェース制御部120と低レベルIOインターフェース制御部1603とを備え、アクセス装置100からいずれのインターフェース制御部を介しても半導体メモリカード110にアクセス可能とする。本実施の形態3と実施の形態2と異なる点は

、ファイルシステムインターフェース制御部120と低レベルIOインターフェース制御部1603により制御される領域が同じファイルシステム管理領域118である点である。以下、本実施の形態における2つのインターフェース制御部120及び1603の機能について説明する。

- [0089] 本実施の形態では、アクセス装置100が2つのインターフェース制御部を経由して同一の領域にアクセスした際に、ファイルシステム内に矛盾が生じないようにする必要がある。そのため、本実施の形態では、基本的にファイルシステムの制御をアクセス装置100側のファイルシステム制御部1701で行い、半導体メモ리카ード100側のファイルシステムインターフェース制御部120では、ファイルシステム内に矛盾が発生しない範囲で、ファイルシステムの制御を補助する処理を実施する。即ち、ファイルシステムインターフェース制御部120は、不揮発性メモリ115上の共通の領域に対してファイルシステムを構築するフォーマット処理のみを実施する。不揮発性メモリ115上の共通の領域に存在するファイルに対するフォーマット処理以外のファイルアクセス処理は、アクセス装置100内のファイルシステム制御部1701から入力されるコマンドに基づいて低レベルIOインターフェース制御部1603が実施する。
- [0090] 本実施の形態におけるファイルシステムインターフェース制御部120の一例として、ファイルシステム管理領域118のフォーマット機能のみを実現する場合について説明する。ファイルシステムのフォーマットは、半導体メモ리카ード110上のデータをファイルシステムで管理するために必要な処理であり、半導体メモ리카ード110を初めて使用する際や、一旦全てのデータを消去する際等を実施される処理である。
- [0091] ここで説明する例では、アクセス装置100はファイルシステムのフォーマットを行う際にのみ、ファイルシステムインターフェース制御部120を介して半導体メモ리카ード110にアクセスする。更に、フォーマット後のファイルアクセスは低レベルIOインターフェース制御部1603を介してアクセスする。このように2つのインターフェース制御部の役割を分担(機能区分)させることで、2つのインターフェース制御部を介して半導体メモ리카ード110にアクセスしても、ファイルシステム内に矛盾が生じないことを保証する。
- [0092] 更に、本実施の形態におけるファイルシステムインターフェース制御部120は、カ

ード情報格納部119から半導体メモリカード110の物理的特性に関するカード情報を取得し、その特性に応じたファイルシステムのフォーマット機能を提供する。第1に、本実施の形態におけるファイルシステムインターフェース制御部120は、カード情報格納部119から取得したカード情報を基に、半導体メモリカード110の最適アクセス単位を決定する。第2に、ファイルシステム管理領域118のフォーマットに際し、マスターブートレコード・パーティションテーブル503からルートディレクトリエントリ507までの管理情報領域501の大きさを調整し、最適アクセス単位の倍数長になるように設定する。

[0093] 次に、本実施の形態におけるフォーマット処理後のデータ配置について、図24を用いて説明する。図24は本実施の形態におけるファイルシステムインターフェース制御部120がファイルシステム管理領域118をフォーマットした場合におけるデータ配置の一例を示した図である。図24においてMBR・PTはマスターブートレコード・パーティションテーブル503、PBSはパーティションブートセクタ504、RDEはルートディレクトリエントリ507を示す。

[0094] 図24に示すようにマスターブートレコード・パーティションテーブル503からルートディレクトリエントリ507までの管理情報領域501の大きさが最適アクセス単位の倍数長となっている。更に、最適アクセス単位がクラスタサイズの倍数長となっている。そのため、アクセス装置100のファイルシステム制御部1701がデータ領域502にクラスタ単位でアクセスした場合に、複数の最適アクセス単位に跨ってアクセスすることがなく、効率的にアクセスすることが可能となる。

[0095] また、アクセス装置100が8つの連続したクラスタにアクセスした場合、半導体メモリカード110内の最適アクセス単位でのアクセスとなり、不揮発性メモリカード115に対する最適なアクセスが実現できる。このようにファイルシステム管理領域118をフォーマットすることにより、アクセス装置100側で半導体メモリカード110の特性を考慮せずにファイルアクセスした場合でも、半導体メモリカード110にとって最適なアクセスに近いアクセスが実施されるようになる。

[0096] 以上説明したように本実施の形態では、半導体メモリカード110内にファイルシステムインターフェース制御部120と、低レベルIOインターフェース制御部1603とを備え

、アクセス装置100からいずれのインターフェース制御部を介しても半導体メモリカードにアクセス可能とする。更に、ファイルシステムインターフェース制御部120は、アクセス装置100のファイルシステム制御部1701が行うファイルシステム制御の補助機能を提供し、2つのインターフェース制御部を介して半導体メモリカード110にアクセスした場合でも、ファイルシステム内に矛盾が発生しないようにする。これにより、アクセス装置100におけるファイルシステム制御の一部を半導体メモリカード110が行い、アクセス装置100の負担を軽減することができる。

[0097] 尚、本実施の形態では、ファイルシステムインターフェース制御部120がファイルシステムのフォーマット機能のみを実現する場合について説明した。本実施の形態の主眼とする点は、半導体メモリカード110のファイルシステム内に矛盾が発生しない範囲で、ファイルシステム制御の補助機能をファイルシステムインターフェース制御部120が実現する点にある。そのため、ファイルシステムインターフェース制御部120はフォーマット機能のみを実現するのではなく、その他のファイルシステム制御の補助機能を実現する構成としても良い。又フォーマット機能を含まずに、その他のファイルシステム制御の補助機能を実現する構成としても良い。例えば、使用するインターフェース制御部を選択するコマンドを半導体メモリカード110に設け、アクセス装置100が明示的に使用するインターフェース制御部を指定して必要に応じて切り替えることで、2つのインターフェース制御部が並列に呼び出されないような構成としても良い。

[0098] また、本実施の形態ではFATファイルシステムを例として説明したが、その他のファイルシステムを使用しても構わない。また、カード情報格納部119に格納されたカード情報は、半導体メモリカード120の使用状態に応じて値を更新可能としても良い。また、カード情報格納部119を含むメモリ116は不揮発性メモリ115に含めても良いし、カード情報を更新しない場合はROM117に含めても良い。

[0099] 以上のように実施の形態3は、半導体メモリカード内にユーザデータ等を格納する不揮発性メモリと、半導体メモリカードの物理的特性に関するカード情報を格納するカード情報格納部と、カード情報格納部に格納された情報を基に半導体メモリカードの物理的特性に適したファイルアクセスを行うファイルシステムインターフェース制御

部と、半導体メモ리카ード外部のアクセス装置から半導体メモ리카ード内の不揮発性メモリに対する低レベル入出力要求を処理する低レベルIOインターフェース制御部とを備え、前記2つのインターフェース制御部からのアクセスにより不揮発性メモリ上に構築されたファイルシステム内に矛盾が生じないように、前記2つのインターフェース制御部の機能を制限することを特徴とする。

[0100] (実施の形態4)

次に本発明の実施の形態4における半導体メモ리카ードについて説明する。図25は実施の形態4における半導体メモ리카ード及びアクセス装置の構成図である。図25において図23の構成と異なる点は、アクセス装置100内のアプリケーションプログラム105からカードインターフェース制御部106を直接呼び出す経路が存在していることと、半導体メモ리카ード110内に同期制御部2501を設けたことである。

[0101] 即ち実施の形態4では、実施の形態3と同様、半導体メモ리카ード110内にファイルシステムインターフェース制御部120と、低レベルIOインターフェース制御部1603とを備え、アクセス装置100からいずれのインターフェース制御部を介しても半導体メモ리카ード110にアクセス可能とする。また、2つのインターフェース制御部を介して不揮発性メモリ115上の同一領域にアクセスする。実施の形態3と異なる点は、半導体メモ리카ード110内に同期制御部2501を備え、2つのインターフェース制御部を介して同一領域をアクセスした場合に、ファイルシステム内に矛盾が生じないように同期をとる点である。

[0102] 本実施の形態のファイルシステムインターフェース制御部120は、カード情報格納部119に格納されたカード情報に基づいて不揮発性メモリ115内に格納したデータをファイルとして管理し、ホストインターフェース部111を介してアクセス装置100から不揮発性メモリ115上のファイルに対するオープン、クローズ、リードの読み出し専用のファイルアクセス処理を要求するコマンドが入力されたとき、不揮発性メモリ115内に存在するファイルに対するファイルアクセス処理を実施する。

[0103] 低レベルIOインターフェース制御部1603は、ファイルシステムインターフェース制御部がデータ読み出しに使用する不揮発性メモリ115内の領域のうち、任意の位置に対するデータ書き込み、又は読み出し処理を要求するコマンドがホストインターフ



エース部111を介してアクセス装置100から入力されたとき、不揮発性メモリ115内の任意の位置に対するデータ書き込み、又は読み出し処理を実施する。

- [0104] 同期制御部2501は低レベルIOインターフェース制御部1603が不揮発性メモリ115内に存在するファイルシステムの管理情報に対するデータ書き込み処理を実施するとき、ファイルシステムインターフェース制御部120が半導体メモリカード110内のRAM上に読み出しているファイルシステム管理情報を更新する。このような構成の場合、本実施の形態のファイルシステムインターフェース制御部120は、実施の形態3のようにフォーマット機能のみ実現する場合に比べ、実現する機能の制約が少なくなる。
- [0105] 続いて本実施の形態における同期制御部2501の機能について、図26～図28を用いて説明する。ここでは、同期制御部2501の例として、ファイルシステムインターフェース制御部120が読み出し専用のファイルシステム機能(上位コマンド)を実現し、同時に低レベルIOインターフェース制御部1603が低レベルIO機能(下位コマンド)を実現する場合について説明する。この場合、アクセス装置100は、ファイルシステム制御部1701によりファイルシステムに対する読み書きを制御し、低レベルIOインターフェース制御部1603を介して半導体メモリカード110の不揮発性メモリ115にアクセスすることが可能である。これと同時にファイルシステムインターフェース制御部120を介して半導体メモリカード110の不揮発性メモリ115上に構築されたファイルシステムに読み出し専用のアクセスを行うことが可能である。
- [0106] 図26は半導体メモリカード110のRAM113上に読み込まれたファイルシステムに関する情報の一例を示した図であり、これらの情報を用いてファイルシステムインターフェース制御部120はファイルシステムを制御する。すなわち、RAM113上には、不揮発性メモリ115から読み込まれたFAT2601やオープン中のファイルに関する情報であるオープンファイル情報(OFI)2602、2603、2604、2605が存在し、ファイルシステムの制御に使用される。
- [0107] 更にRAM113上には、同期制御部2501が使用する情報としてキャッシュ管理テーブル2606が存在する。図27はキャッシュ管理テーブル2606に含まれる情報を示した図である。キャッシュ管理テーブル2606は、FAT2601やオープンファイル情報

2602、2603、2604、2605等、RAM113上に読み込まれている情報の不揮発性メモリ115上の位置を示す情報である。図26及び図27の例では、不揮発性メモリ115上の234セクタの位置から始まる123セクタの領域に存在するFAT2601がRAM113上に読み込まれている。また、480セクタの位置にディレクトリエントリ(DE)を持つファイルが2つと、513セクタの位置にディレクトリエントリを持つファイルが1つオープンされており、RAM113上にオープンファイル情報2602、2603、2604、2605がキャッシュされている。

- [0108] 次に図28を用いて、本実施の形態における同期制御部2501の処理について説明する。本実施の形態において、アクセス装置100から半導体メモリカード110に発行されるコマンドは一旦同期制御部2501が受信し、ファイルシステムインターフェース制御部120、低レベルIOインターフェース制御部1603に夫々上位コマンド、下位コマンドを振り分ける。図28はこのような同期制御部2501の処理手順を示した図である。
- [0109] 同期制御部2501の処理において、まずアクセス装置100からコマンドを受信する(S2801)。次に、受信したコマンドを参照し、自身が認識できない不正コマンドか否かを判定する(S2802)。不正コマンドの場合はアクセス装置100にエラーを通知して処理を終了する(S2803)。認識可能なコマンドの場合、そのコマンドが図18で説明したRAW\_WRITEコマンドであるか判定する(S2804)。RAW\_WRITEコマンド以外の場合、コマンドの種類を判別する(S2805)。上位コマンドであればファイルシステムインターフェース制御部120を呼び出し、下位コマンドであれば低レベルIOインターフェース制御部1603を呼び出し、夫々のコマンドに対応した他の処理を実施し、処理を終了する(S2806)。
- [0110] RAW\_WRITEコマンドの場合、コマンドの引数で指定された書き込み位置がRAM113上に読み込んでいるFAT2601と同一であるか判定する(S2807)。図26及び図27の例では、FAT2601は234セクタの位置から始まる123セクタの領域に存在するため、RAW\_WRITEコマンドで例えば256セクタから始まる32セクタのデータ書き込みを要求された場合等は、書き込み位置が同一であると判定する。
- [0111] 書き込み位置が同一であると判定した場合、RAM113上のFAT2601を、RAW

\_\_WRITEコマンドで送信されたデータで更新する(S2808)。そして低レベルIOインターフェース制御部1603を呼び出してRAW\_\_WRITEコマンドを実施し、処理を終了する(S2809)。書き込み位置が同一でないと判定した場合、オープン中のディレクトリエントリ(DE)と同じセクタへの書き込みであるか判定する(S2810)。

- [0112] 図26及び図27の例では、480セクタ、513セクタに存在するディレクトリエントリを持つファイルがオープンされているため、RAW\_\_WRITEコマンドでこれら2セクタのいずれかの位置に対するデータ書き込みを要求された場合、書き込み位置が同一であると判定する。書き込み位置が同一でないと判定した場合、S2813の処理に進む。書き込み位置が同一であると判定した場合、RAW\_\_WRITEコマンドで送信されたデータを参照し、ディレクトリエントリを含むセクタに対する書き込みが、オープン中のファイルのディレクトリエントリが変化するようなデータであるかを判定する(S2811)。
- [0113] ここでディレクトリエントリが変化する場合には、例えばファイルサイズやタイムスタンプ、ファイル名が変更された場合、ディレクトリエントリ自体が消去された場合等が存在する。ディレクトリエントリが変化しないと判定した場合、S2813の処理に進む。ディレクトリエントリが変化すると判定した場合、RAM113上のオープンファイル情報(O FI) 2602、2603、2604、2605を更新する(S2812)。
- [0114] 例えば、ファイルサイズやタイムスタンプ、ファイル名が変更された場合は、オープンファイル情報2602、2603、2604、2605内の各値を更新する。ディレクトリエントリ自体が消去された場合は、オープンファイル情報2602、2603、2604、2605をクリアし、ファイルがオープンされていない状態に更新する。最後に、低レベルIOインターフェース制御部1603を呼び出してRAW\_\_WRITEコマンドを実施し、処理を終了する(S2813)。
- [0115] このように、同期制御部2501は、半導体メモ리카ード110に対する書き込みアクセスの際に書き込み位置を確認し、RAM113上に読み込まれたデータが変化するような書き込みの場合には、データの書き込みと同時にRAM113上のデータを更新する。これにより、ファイルシステムインターフェース制御部120が読み出し専用のファイルシステム機能を実現する場合に、低レベルIOインターフェース制御部1603を介し

での書き込み処理に同期し、ファイルシステム管理情報を送信することにより、ファイルシステム内に矛盾を生じさせることなく処理を行うことが可能となる。

- [0116] 又本実施の形態では、半導体メモ리카ード111内でファイルシステムが存在し論理的にクラスタが連続している場合には、最適アドレス単位の1単位について1回のアドレス変換を実施するだけでよく、アドレス変換処理のオーバーヘッドを低減することができる。
- [0117] 以上説明したように、本実施の形態では、半導体メモ리카ード110内にファイルシステムインターフェース制御部120と、低レベルIOインターフェース制御部1603とを備え、アクセス装置100からいずれのインターフェース制御部を介しても半導体メモ리카ード110にアクセス可能とする。また同期制御部2501を備えることにより、2つのインターフェース制御部を介したアクセス間の同期をとることができる。これにより、2つのインターフェース制御部を介したアクセスにより、ファイルシステム内に矛盾が生じる機会を無くし、実施の形態3よりもファイルシステムインターフェース制御部120が実現する機能を拡大することができる。
- [0118] 尚、本実施の形態では、ファイルシステムインターフェース制御部120が読み出し専用のファイルシステム機能を実現する場合について説明した。本実施の形態の主眼とする点は、半導体メモ리카ード110が2つのインターフェース制御部を備え、各々のインターフェース制御部を介してアクセスされた場合に、ファイルシステム内に矛盾が生じないように同期制御部2501で同期をとる点にある。そのため、ファイルシステムインターフェース制御部120が読み出し専用のファイルシステム機能を実現するのは一例であり、同期制御部2501においてファイルシステム内に矛盾が生じないように同期をとることが可能な場合は、その他の機能を実現しても良い。
- [0119] また、本実施の形態ではFATファイルシステムを例として説明したが、その他のファイルシステムを使用しても構わない。また、カード情報格納部119に格納されたカード情報は、半導体メモ리카ード120の使用状態に応じて値を更新可能としても良い。また、カード情報格納部119を含むメモリ116は不揮発性メモリ115に含めても良いし、カード情報を更新しない場合はROM117に含めても良い。
- [0120] 以上のように実施の形態4は、半導体メモ리카ード内にユーザデータ等を格納する

不揮発性メモリと、半導体メモ리카ードの物理的特性に関するカード情報を格納するカード情報格納部と、カード情報格納部に格納された情報を基に半導体メモ리카ードの物理的特性に適したファイルアクセスを行うファイルシステムインターフェース制御部と、半導体メモ리카ード外部のアクセス装置から半導体メモ리카ード内の不揮発性メモリに対する低レベル入出力要求を処理する低レベルIOインターフェース制御部と、ファイルシステムインターフェース制御部及び低レベルIOインターフェース制御部からの不揮発性メモリに対する処理を同期させる同期制御部とを備えることを特徴とする。

[0121] (実施の形態5)

次に本発明の実施の形態5における半導体メモ리카ードについて説明する。図29は本実施の形態における半導体メモ리카ード110を中心とする構成図である。図29に示す半導体メモ리카ード110が図1の構成と異なる点は、半導体メモ리카ード110内に複数のファイルシステムインターフェース制御部(A, B, C)2901、2902、2903が存在する点と、カード情報格納部119内にファイルシステム種別フラグ2904が存在する点である。

[0122] 即ち本実施の形態では、ファイルシステムインターフェース制御部2901、2902、2903は、管理するファイルシステムの種別が夫々異なる。カード情報格納部119は、不揮発性メモリ115の消去ブロックサイズを含む半導体メモ리카ードの物理的特性に関する情報、及び不揮発性メモリ115上に構築されているファイルシステムの種別を示すファイルシステム種別フラグを含むカード情報を格納する。

[0123] ファイルシステムインターフェース制御部2901〜2903は、カード情報格納部119に格納されたカード情報に基づいて不揮発性メモリ115内に格納したデータをファイルとして管理し、ホストインターフェース部111を介してアクセス装置100から入力されたコマンドに基づいて不揮発性メモリ115上のファイルに対するオープン、クローズ、リード、ライトを含むファイルアクセス処理を実施する。そして複数のファイルシステムインターフェース制御部2901〜2903のうち、ファイルシステム種別フラグに対応するファイルシステムインターフェース制御部が半導体メモ리카ード115上で動作する。

- [0124] アクセス装置100からはいずれのインターフェース制御部を介しても半導体メモリカード110にアクセス可能とする。そのため、半導体メモリカード110内の不揮発性メモリ115を管理するファイルシステムの種別を、用途に応じて変更することが可能となる。
- [0125] 続いて本実施の形態におけるファイルシステムインターフェース制御部2901～2903について説明する。ファイルシステムインターフェース制御部2901～2903は夫々異なる種類のファイルシステムを解釈してアクセス制御する。本実施の形態ではファイルシステムのフォーマット時に、使用するファイルシステムの種別を決定し、カード情報格納部119にファイルシステム種別フラグ(図中では単に種別フラグ)2904を保持する。ファイルシステム種別フラグ2904は選択したファイルシステムの種類、又は使用するファイルシステムインターフェース制御部2901、2902、2903を一意に特定することが可能な情報である。以降、インターフェース制御部2901、2902、2903を介して不揮発性メモリ115にアクセスする際には、ファイルシステム種別フラグ2904を参照し、3つのインターフェース制御部2901、2902、2903から使用するインターフェース制御部を選択して、ファイルシステムの制御を行う。
- [0126] 図30は、本実施の形態におけるファイルシステムインターフェース制御部2901、2902、2903の選択手順を示したフローチャートである。図30の処理において、まずアクセス装置100からコマンドを受信する(S3001)。次に、受信したコマンドを参照し、自身が認識できない不正コマンドか否かを判定する(S3002)。不正コマンドの場合、アクセス装置100にエラーを通知して処理を終了する(S3003)。認識可能なコマンドの場合、そのコマンドがFORMATコマンドであるか判定する(S3004)。FORMATコマンド以外の場合、S3010の処理に進む。FORMATコマンドの場合、コマンドの引数が正しい値かを判定する(S3005)。
- [0127] 指定された引数が解釈不可能であった場合等、不正な引数が指定された場合、エラーが発生したと判定し、アクセス装置100にエラーを通知して処理を終了する(S3006)。引数が正しい場合、引数を基に使用するファイルシステムインターフェース制御部2901、2902、2903を選択する(S3007)。ここで引数では、使用するファイルシステムの種類(FAT、UDF等)を直接指定するか、あるいは“ファイルシステムイン

ターフェース制御部A”等のインターフェース制御部を一意に特定するフラグを指定する。こうすることにより、半導体メモリカード110が使用するファイルシステムインターフェース制御部2901、2902、2903を選択できるようにする。

[0128] 次に、ファイルシステム種別フラグ2904をカード情報格納部119内に設定する(S3008)。次に、選択したファイルシステムインターフェース制御部2901、2902、2903を呼び出し、フォーマット処理を実施し、S3014の処理に進む(S3009)。この処理により呼び出されたファイルシステムインターフェース制御部2901、2902、2903は、自身が制御するファイルシステム種別に応じて、不揮発性メモリ115をフォーマットする。

[0129] 一方、S3004の処理においてFORMATコマンド以外であると判定した場合、コマンドの引数が正しい値かを判定する(S3010)。指定された引数が解釈不可能であった場合等、不正な引数が指定された場合はエラーが発生したと判定し、アクセス装置100にエラーを通知して処理を終了する(S3011)。引数が正しい場合、ファイルシステム種別フラグ2904を参照し、使用するファイルシステムインターフェース制御部2901、2902、2903を選択する(S3012)。

[0130] 次に、選択したファイルシステムインターフェース制御部2901、2902、2903を呼び出し、各コマンドの処理を実施する(S3013)。この処理により呼び出されたファイルシステムインターフェース制御部2901、2902、2903は、自身が制御するファイルシステム種別に応じて、ファイルオープンやファイルデータの読み出し等の処理を実施する。次に、各ファイルシステムインターフェース制御部2901、2902、2903で実施した処理が成功したかを判定する(S3014)。処理に失敗した場合、アクセス装置100にエラーを通知して処理を終了する(S3015)。処理に成功した場合、アクセス装置100に処理完了を通知して処理を終了する(S3016)。

[0131] 以上説明したように本実施の形態では、管理するファイルシステムの種別が異なる複数のファイルシステムインターフェース制御部2901、2902、2903を半導体メモリカード110内に備え、アクセス装置100からいずれのインターフェース制御部を介しても半導体メモリカード110にアクセス可能とする。これにより、大容量ファイルへの高速アクセスに適したファイルシステムや、数多くの小容量ファイルをアクセスするのに

適したファイルシステム等を用意することができる。そして半導体メモリカード110の使用用途に応じてファイルシステムの種別を変更することで、各用途に適したファイルシステムを使用することができる。

- [0132] 尚、本実施の形態ではファイルシステムインターフェース制御部2901、2902、2903が3種類、半導体メモリカード110内に存在する例を用いて説明した。しかし3種類に限らず、任意の複数種類で構成しても良い。また本実施の形態では、実施の形態1を基本にして相違点を説明したが、本実施の形態の発明内容を、先に説明したその他の実施の形態と組み合わせて使用しても良く、低レベルIOインターフェース制御部や同期制御部を含む構成としても良い。
- [0133] また、本実施の形態ではファイルシステムインターフェース制御部2901、2902、2903がROM117に含まれる構成について説明した。しかし、メモリ116や、不揮発性メモリ115に含まれる構成とし、ファイルシステムインターフェース制御部2901、2902、2903を更新可能な構成としても良い。即ち半導体メモリカード110の外部から、新たなファイルシステムの種別に対応したファイルシステムインターフェース制御部を追加したり、既存のファイルシステムインターフェース制御部を更新したり、不要なファイルシステムインターフェース制御部を削除可能な構成としても良い。
- [0134] また、ファイルシステムインターフェース制御部で管理するファイルシステムは、FATファイルシステムやUDFファイルシステム以外の任意のファイルシステムを使用しても良い。また、カード情報格納部119に格納されたカード情報は、半導体メモリカード110の使用状態に応じて値を更新可能としても良い。また、カード情報格納部119を含むメモリ116は、不揮発性メモリ115に含めても良い。
- [0135] 以上のように実施の形態5は、半導体メモリカード内にユーザデータ等を格納する不揮発性メモリと、半導体メモリカードの物理的特性に関するカード情報を格納するカード情報格納部と、複数種類のファイルシステムに対応し、カード情報格納部に格納された情報を基に半導体メモリカードの物理的特性に適したファイルアクセスを行う複数のファイルシステムインターフェース制御部とを備えることを特徴とする。
- [0136] 尚、前述した各実施の形態では、半導体メモリカードについて説明しているが、本発明はカード状の半導体メモリに限らず、他の種々の形状の半導体メモリ装置に適



用することができる。

#### 産業上の利用可能性

- [0137] 本発明に関わる半導体メモリ装置は、半導体メモリ装置の特性に関する情報を格納する装置情報格納部と、その情報を基に半導体メモリ装置の特性に適したファイルアクセスを行うインターフェース制御部を半導体メモリ装置内に備えることにより、アクセス装置が半導体メモリ装置の特性を意識することなく、最適なファイルアクセスを実現することができる。このような半導体メモリ装置は、デジタルAV機器や携帯電話端末、PC等の情報記録媒体として利用することができる。また、半導体メモリ装置の特性に応じた最適なファイルアクセスを実現するため、転送レートの高い高品質AVデータを記録する機器の情報記録媒体として利用する場合は特に好適に機能する。

### 請求の範囲

- [1] 複数のセクタからなり、前記複数のセクタのうち連続する特定個のセクタがデータ消去の最小単位であるブロックとしてグループ化され、外部のアクセス装置から送られたデータを書き込み、及び読み出す不揮発性メモリと、
- 前記アクセス装置から制御信号を含むコマンドが入力されたとき、前記不揮発性メモリに対するデータの消去、書き込み、及び読み出しを制御するメモリコントローラと、
- 前記不揮発性メモリの消去ブロックサイズを含む半導体メモリ装置の物理的特性に関する装置情報を格納する装置情報格納部と、
- 前記装置情報格納部に格納された装置情報に基づいて前記不揮発性メモリに対してファイルアクセス処理を実施するファイルシステムインターフェース制御部と、を具備する半導体メモリ装置。
- [2] 前記ファイルシステムインターフェース制御部は、前記不揮発性メモリに格納したデータをファイルとして管理し、前記アクセス装置から前記不揮発性メモリ上のファイルに対するファイルアクセス処理を要求するコマンドが入力されたとき、前記不揮発性メモリ内に存在するファイルに対するファイルアクセス処理を実施する請求項1記載の半導体メモリ装置。
- [3] 前記不揮発性メモリは、
- 第1の領域と、第2の領域と、を有するものであり、
- 前記ファイルシステムインターフェース制御部は、前記不揮発性メモリの前記第1の領域に格納したデータをファイルとして管理し、前記アクセス装置から前記不揮発性メモリの前記第1の領域のファイルに対するファイルアクセス処理を要求するコマンドが入力されたとき、前記不揮発性メモリの前記第1の領域に存在するファイルに対するファイルアクセス処理を実施するものであり、
- 前記半導体メモリ装置は、ファイルシステムの制御を行う前記アクセス装置から前記不揮発性メモリの前記第2の領域に対するデータの書き込み、又は読み出し処理を要求するコマンドが入力されたとき、前記不揮発性メモリ内の前記第2の領域に対するデータ書き込み、又は読み出し処理を実施する低レベルIOインターフェース制御部を更に具備する請求項1記載の半導体メモリ装置。

- [4] 半導体メモリ装置は、ファイルシステムの制御を行う前記アクセス装置から前記不揮発性メモリの任意の位置に対するデータの書き込み、又は読み出し処理を要求するコマンドが入力されたとき、前記不揮発性メモリ内の任意の位置に対するデータ書き込み、又は読み出し処理を実施する低レベルIOインターフェース制御部を更に具備し、

前記不揮発性メモリは、前記ファイルシステムインターフェース制御部と前記低レベルIOインターフェース制御部の両方により制御される共通の領域を有し、

前記ファイルシステムインターフェース制御部は、前記不揮発性メモリ上の共通の領域に対して前記装置情報格納部に格納された装置情報に基づいてアクセス単位を最適のアクセス単位となるようにファイルシステムを構築するフォーマット処理のみを実施し、

前記不揮発性メモリ上の共通の領域に存在するファイルに対するフォーマット処理以外のファイルアクセス処理は、前記アクセス装置から入力されるコマンドに基づいて前記低レベルIOインターフェース制御部が実施する請求項1記載の半導体メモリ装置。

- [5] 前記ファイルシステムインターフェース制御部は、前記アクセス装置から前記不揮発性メモリ上のファイルに対する読み出し専用のファイルアクセス処理を要求するコマンドが入力されたとき、前記不揮発性メモリ内に存在するファイルに対するファイルアクセス処理を実施するものであり、

前記半導体メモリ装置は、

前記ファイルシステムインターフェース制御部がデータ読み出しに使用する前記不揮発性メモリ内の領域のうち、任意の位置に対するデータ書き込み、又は読み出し処理を要求するコマンドがファイルシステムの制御を行う前記アクセス装置から入力されたとき、前記不揮発性メモリ内の任意の位置に対するデータ書き込み、又は読み出し処理を実施する低レベルIOインターフェース制御部と、

前記低レベルIOインターフェース制御部が前記不揮発性メモリ内に存在するファイルシステムの管理情報に対するデータ書き込み処理を実施するとき、前記ファイルシステムインターフェース制御部が前記半導体メモリ装置内の一時記憶メモリ上に読み

出しているファイルシステム管理情報を矛盾が生じないように更新する同期制御部と、を更に有する請求項1記載の半導体メモリ装置。

- [6] 前記装置情報格納部は、前記不揮発性メモリの消去ブロックサイズを含む半導体メモリ装置の物理的特性に関する情報、及び前記不揮発性メモリ上に構築されているファイルシステムの種別を示すファイルシステム種別フラグを含む装置情報を格納するものであり、

前記ファイルシステムインターフェース制御部は、前記装置情報格納部に格納された装置情報に基づいて前記不揮発性メモリ内に格納したデータをファイルとして管理し、前記アクセス装置から入力されたコマンドに基づいて前記不揮発性メモリ上のファイルに対するファイルアクセス処理を実施する複数のファイルシステムインターフェース制御部であり、

前記複数のファイルシステムインターフェース制御部のうち、前記ファイルシステム種別フラグに対応する前記ファイルシステムインターフェース制御部が前記半導体メモリ装置上で動作する請求項1記載の半導体メモリ装置。

- [7] 前記ファイルシステムインターフェース制御部は、

前記装置情報格納部に格納された消去ブロックサイズを含む情報に基づき前記不揮発性メモリへのアクセス単位を決定し、前記不揮発性メモリ上の領域にファイルシステムを構築するフォーマット処理を実施する際に、ファイルシステムの管理情報領域の大きさを前記アクセス単位の倍数長に設定する請求項1記載の半導体メモリ装置。

- [8] 前記ファイルシステムインターフェース制御部は、

前記装置情報格納部に格納された消去ブロックサイズを含む情報に基づき前記不揮発性メモリへの最適アクセス単位を決定し、ファイルデータを前記半導体メモリ装置に記録する際の領域割り当て単位として前記最適アクセス単位を使用する請求項1記載の半導体メモリ装置。

- [9] 前記ファイルシステムインターフェース制御部は、

前記装置情報格納部に格納された消去ブロックサイズを含む情報に基づき前記不揮発性メモリへの最適アクセス単位を決定し、複数のディレクトリ領域が同一の前記

最適アクセス単位内に含まれるようにディレクトリ領域を割り当てる請求項1記載の半導体メモリ装置。

- [10] 前記ファイルシステムインターフェース制御部は、  
更新可能な不揮発性の記録媒体であるメモリに格納されたプログラムによって構成され、前記半導体メモリ装置の外部から前記ファイルシステムインターフェース制御部を入れ替え、更新、若しくは削除できるようにした請求項1記載の半導体メモリ装置。
- [11] 複数のセクタからなり、前記複数のセクタのうち連続する特定個のセクタがデータ消去の最小単位であるブロックとしてグループ化された不揮発性メモリを有する半導体メモリ装置の制御方法であって、  
装置情報格納部に前記不揮発性メモリの消去ブロックサイズを含む半導体メモリ装置の物理的特性に関する装置情報をあらかじめ格納し、  
外部のアクセス装置より制御信号を含むコマンドを受け、  
ファイルシステムインターフェース制御部により前記装置情報格納部に格納された装置情報に基づいて前記不揮発性メモリ内に対してファイルアクセス処理を実施し、  
受付けたコマンドに基づいて前記不揮発性メモリに対するデータの消去、書き込み、及び読み出しを行う半導体メモリ装置の制御方法。
- [12] 前記ファイルシステムインターフェース制御部により、前記不揮発性メモリに格納したデータをファイルとして管理し、  
前記アクセス装置から前記不揮発性メモリ上のファイルに対するファイルアクセス処理を要求するコマンドが入力されたとき、前記不揮発性メモリ内に存在するファイルに対するファイルアクセス処理を実施する請求項11記載の半導体メモリ装置の制御方法。
- [13] 前記ファイルシステムインターフェース制御部により、前記装置情報格納部に格納された消去ブロックサイズを含む情報に基づき前記不揮発性メモリへのアクセス単位を決定し、  
前記不揮発性メモリ上の領域にファイルシステムを構築するフォーマット処理を実施する際に、ファイルシステムの管理情報領域の大きさを前記アクセス単位の倍数長

に設定する請求項11記載の半導体メモリ装置の制御方法。

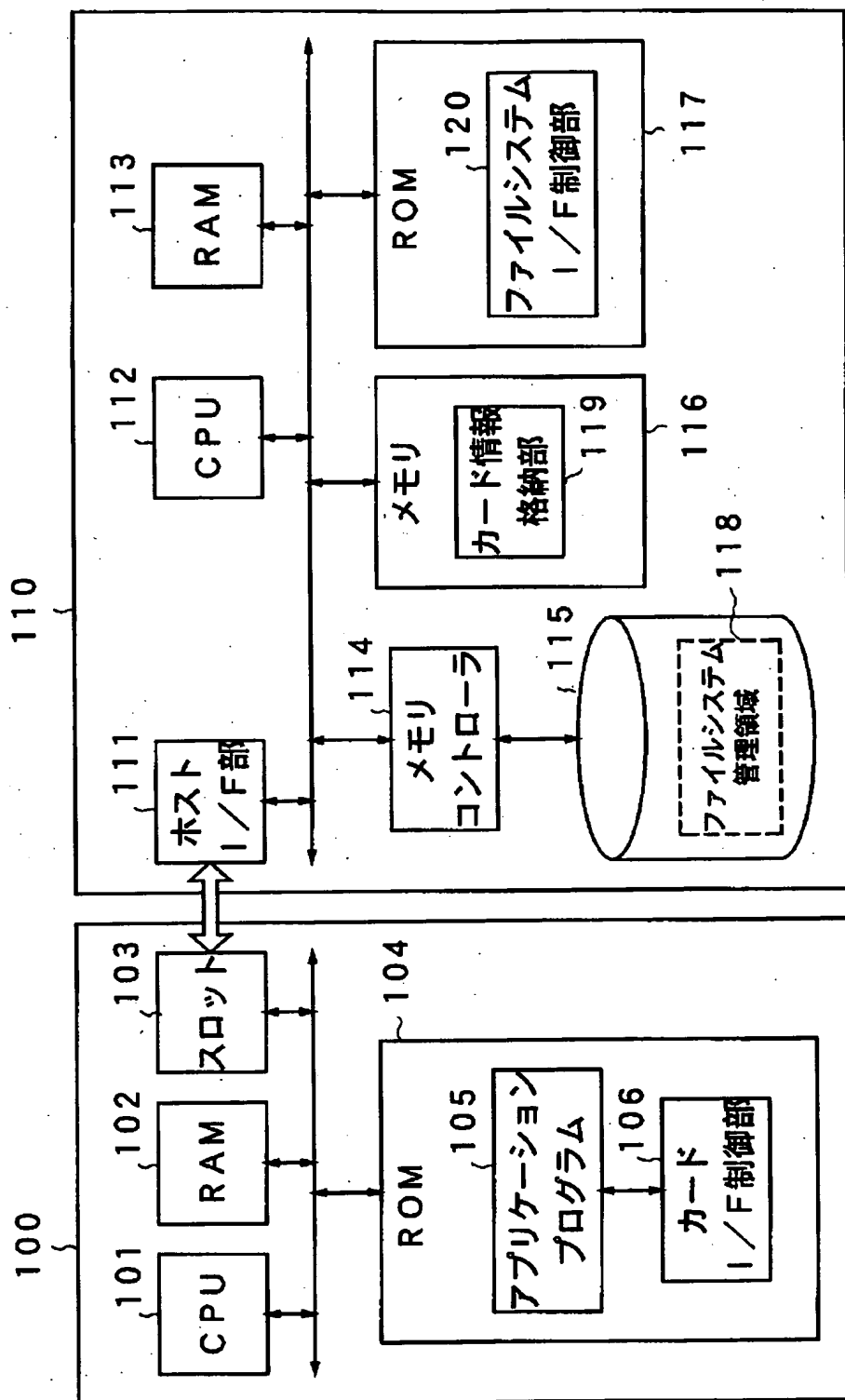
- [14] 前記ファイルシステムインターフェース制御部により、前記装置情報格納部に格納された消去ブロックサイズを含む情報に基づき前記不揮発性メモリへの最適アクセス単位を決定し、

ファイルデータを前記半導体メモリ装置に記録する際の領域割り当て単位として前記最適アクセス単位を使用する請求項11記載の半導体メモリ装置の制御方法。

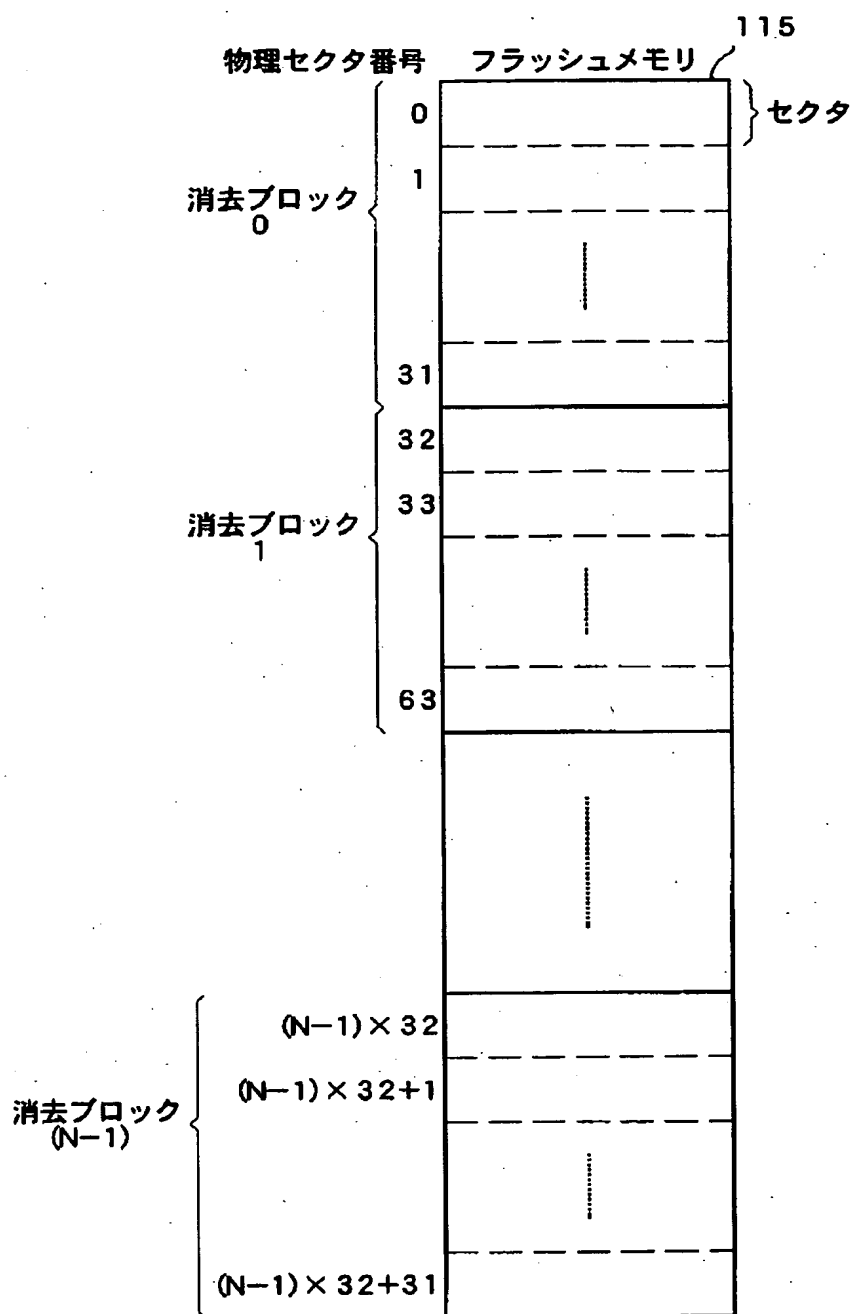
- [15] 前記ファイルシステムインターフェース制御部により、前記装置情報格納部に格納された消去ブロックサイズを含む情報に基づき前記不揮発性メモリへの最適アクセス単位を決定し、

複数のディレクト領域が同一の前記最適アクセス単位内に含まれるようにディレクトリ領域を割り当てる請求項11記載の半導体メモリ装置の制御方法。

[図1]

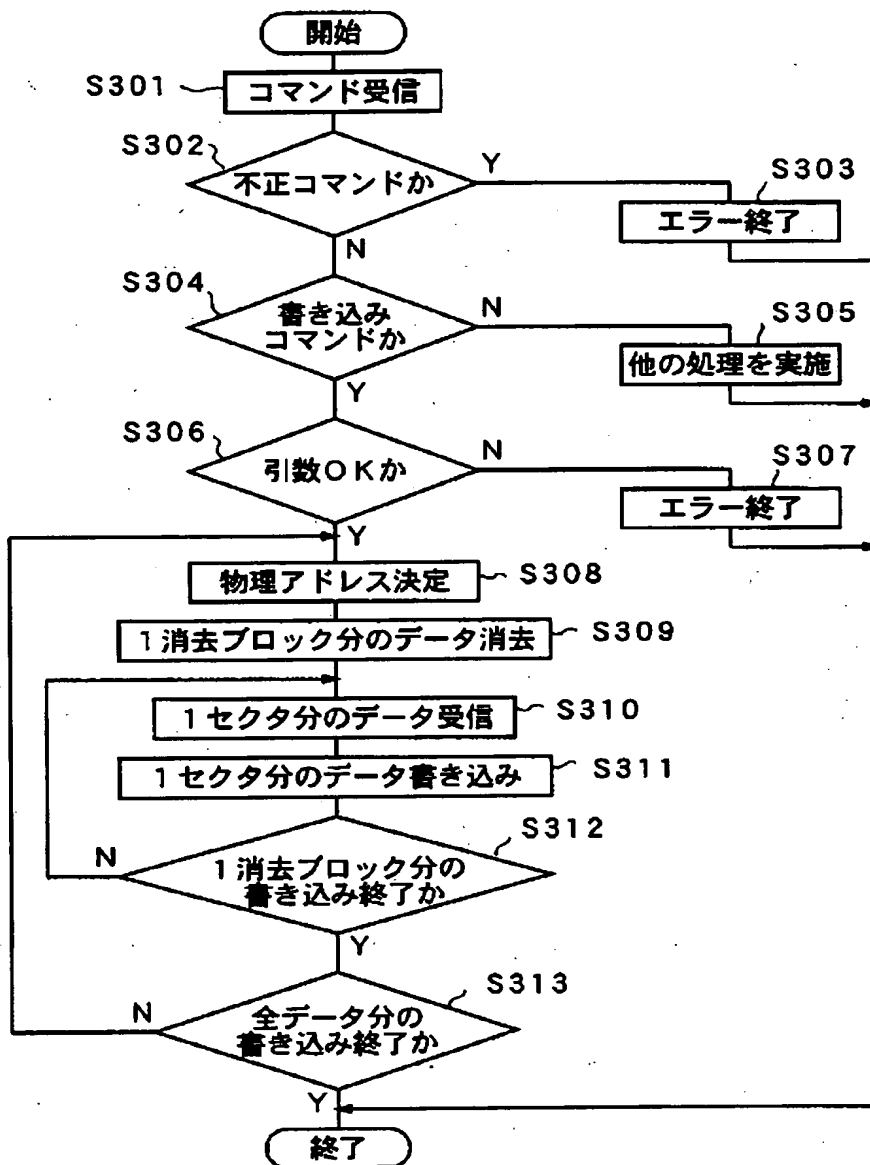


[図2]

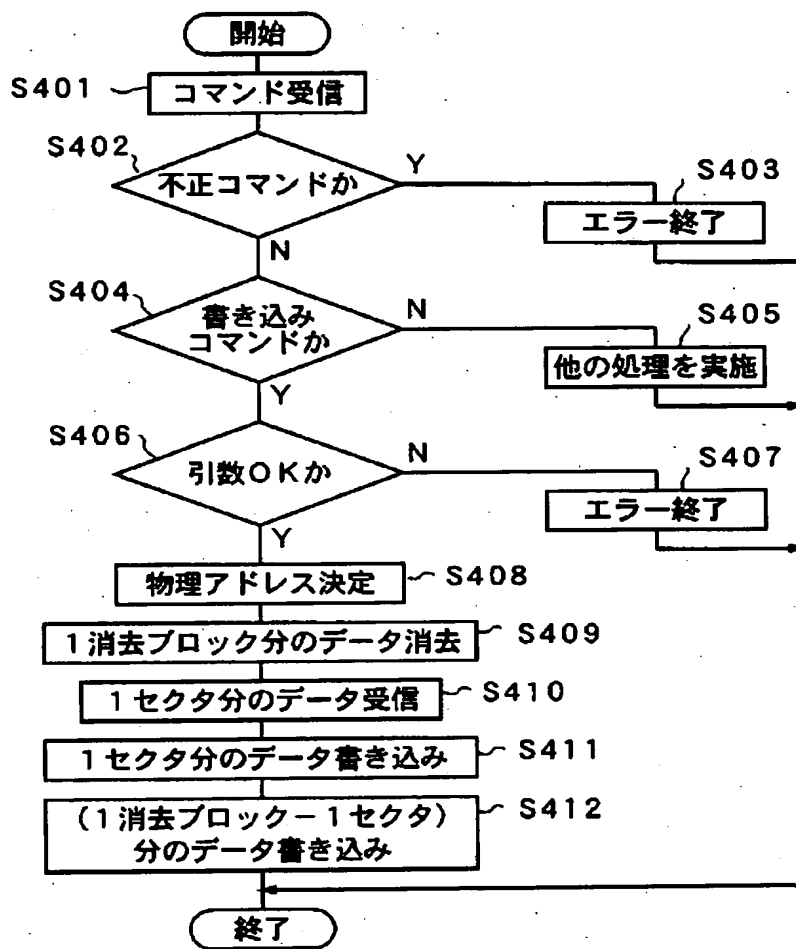




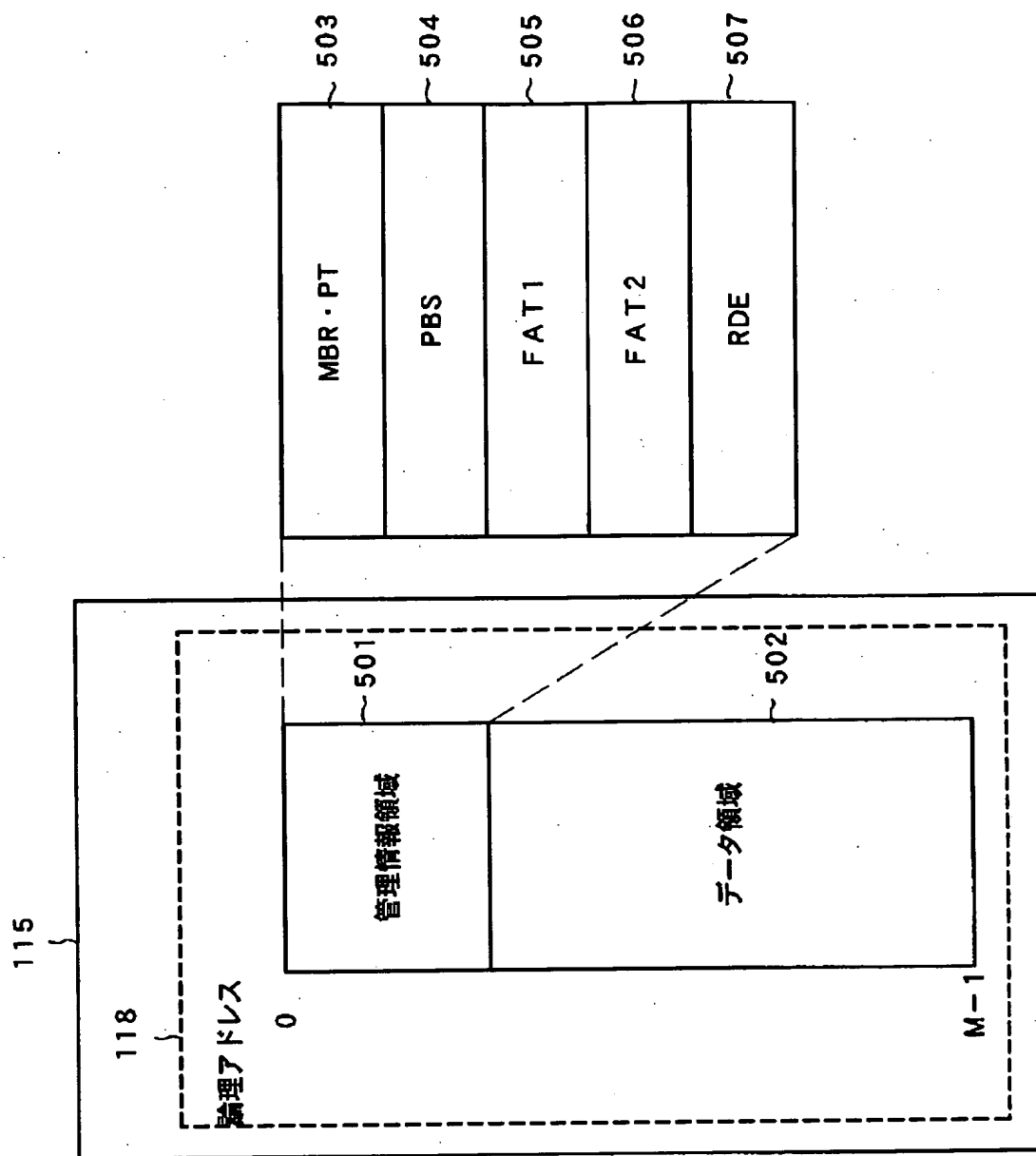
[図3]



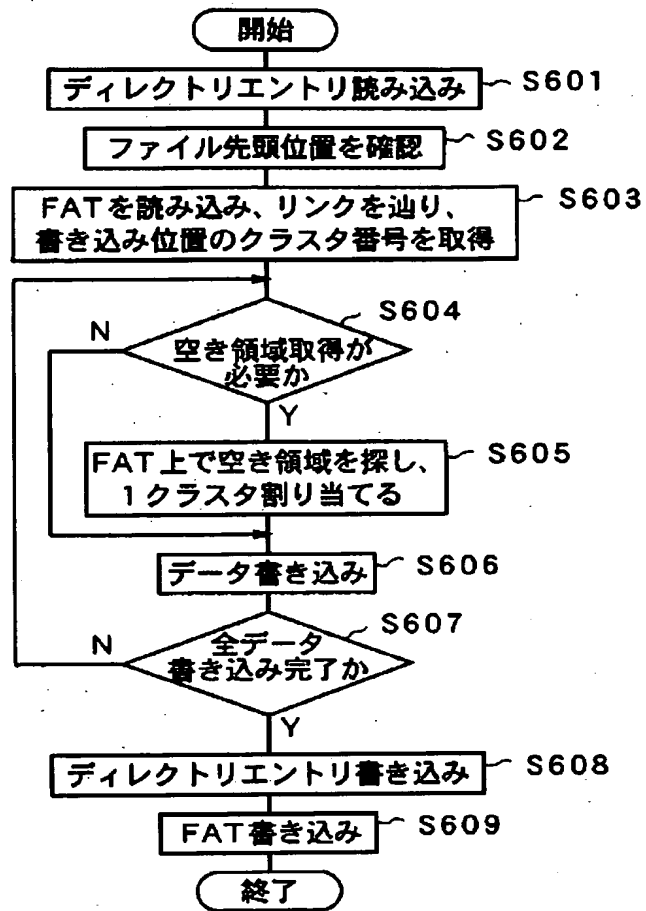
[図4]



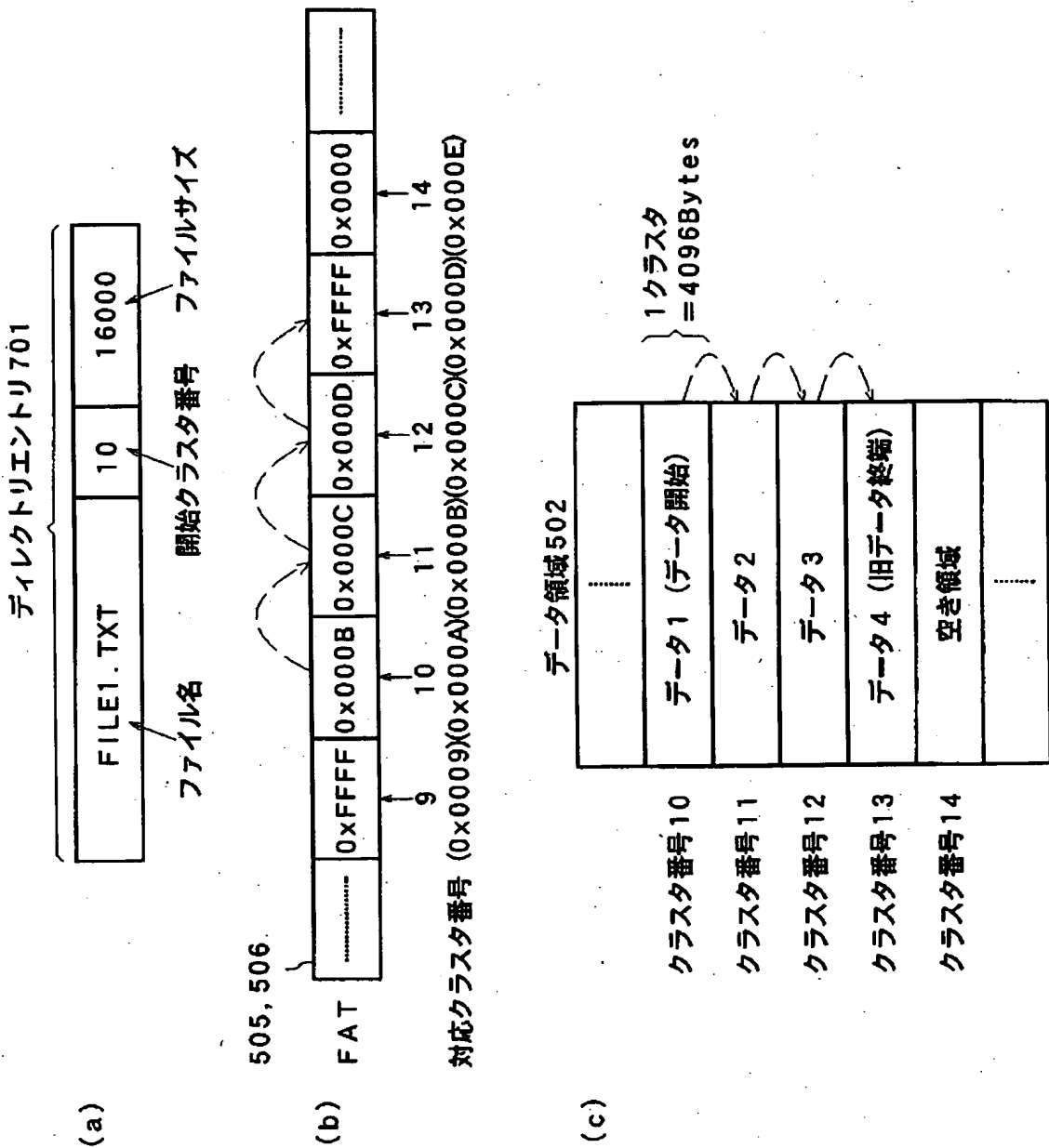
[図5]



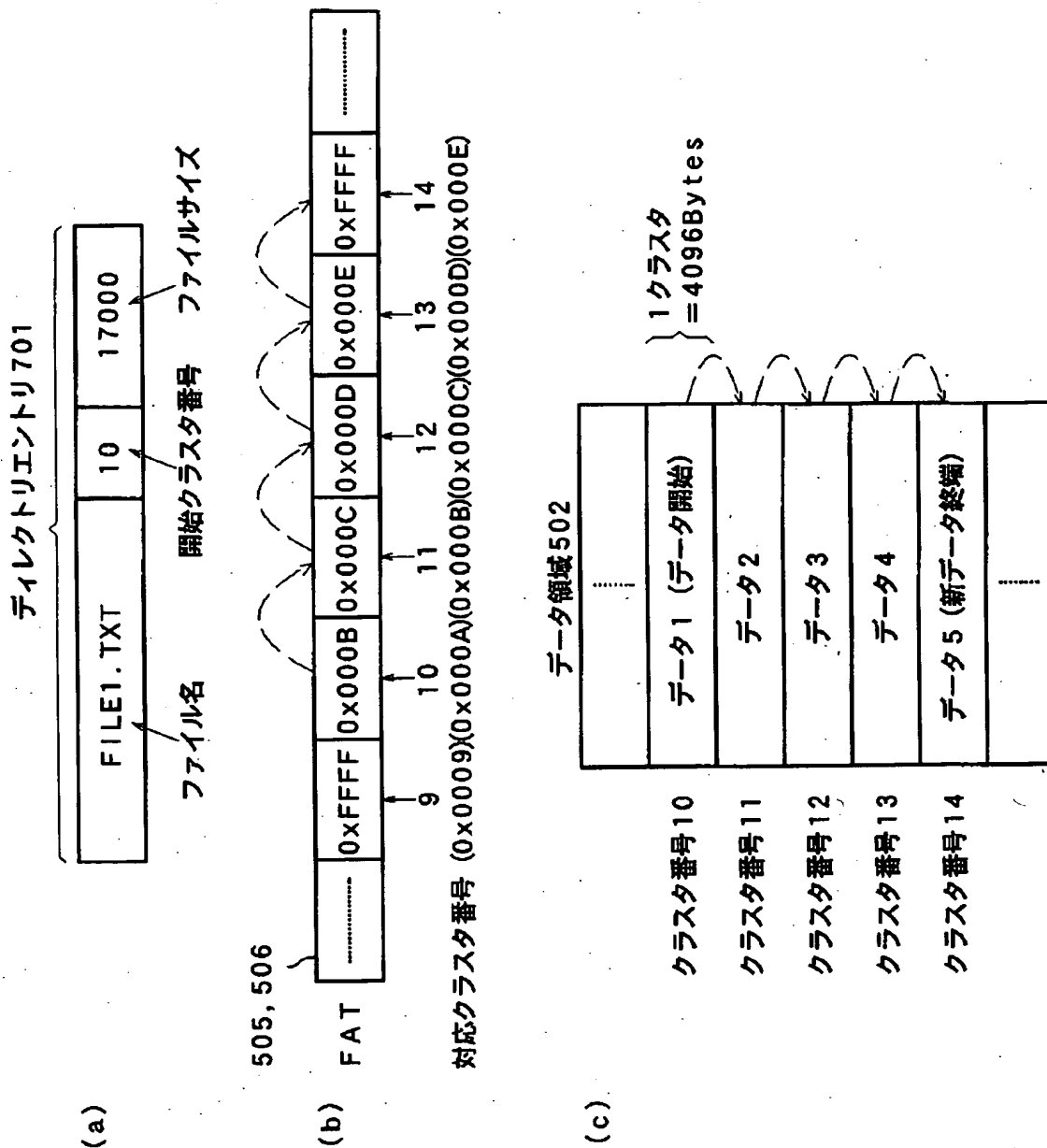
[図6]



[図7]



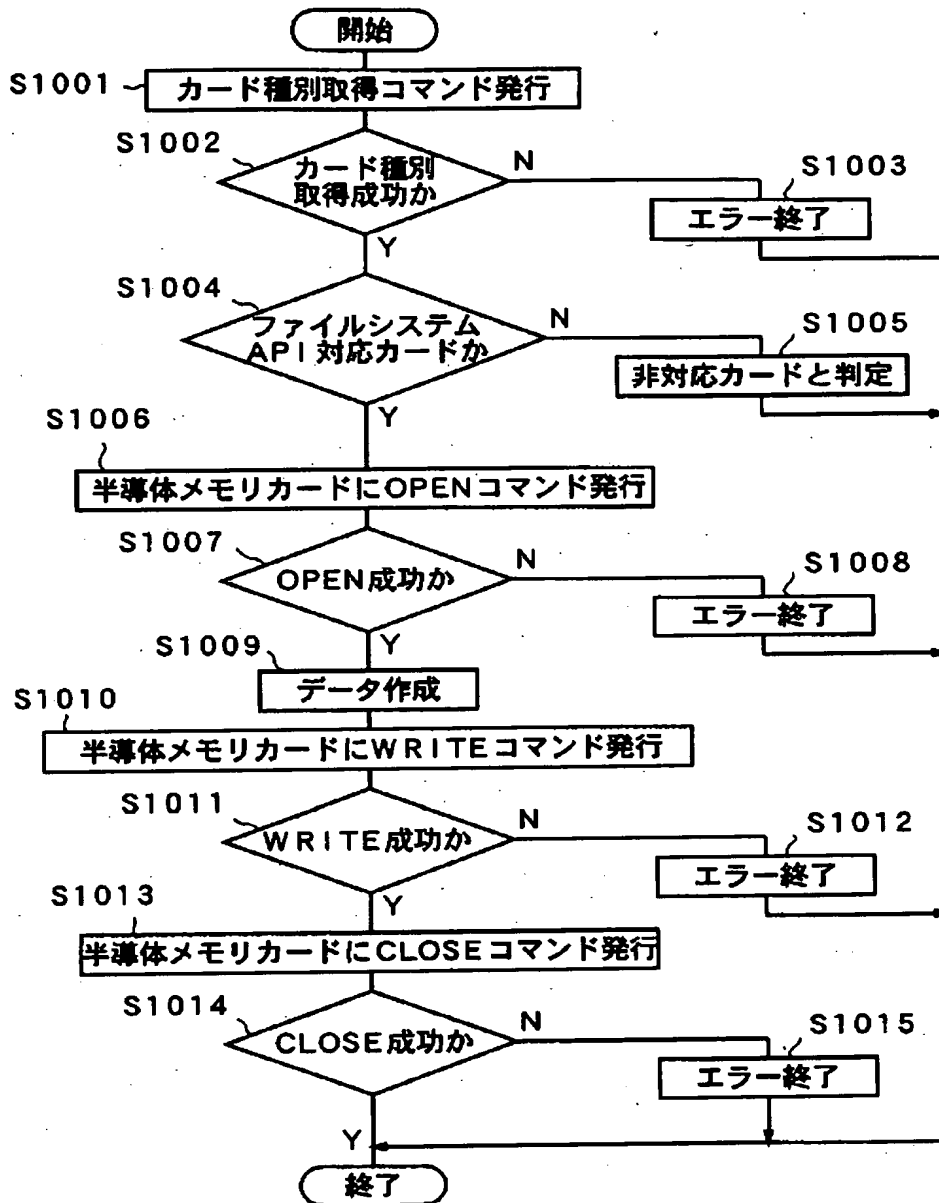
[図8]



[図9]

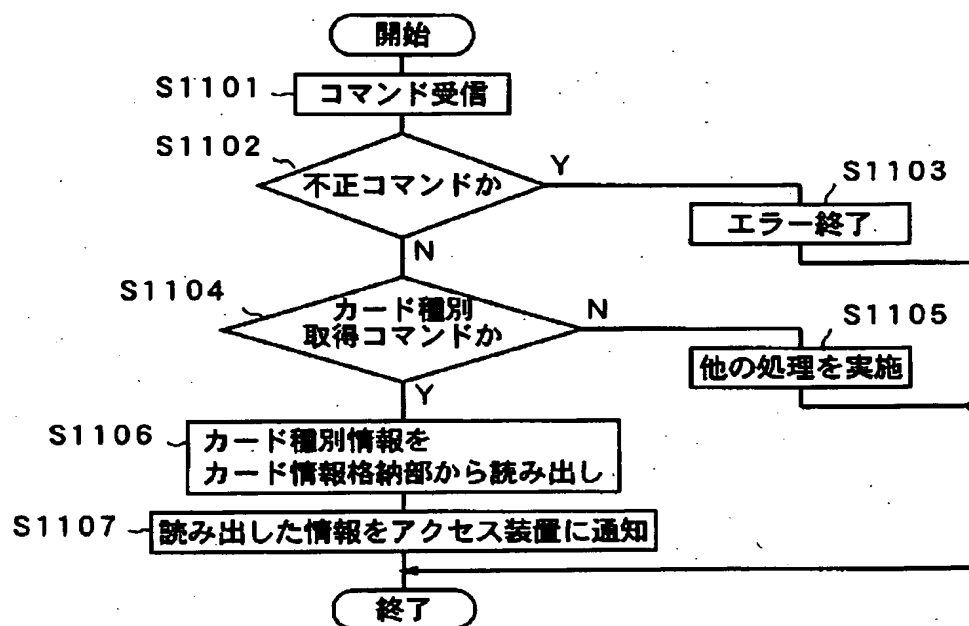
ファイルシステムAPI	項目
OPEN	ファイルをオープンする
CLOSE	ファイルをクローズする
READ	ファイルからデータを読み込む
WRITE	ファイルへデータを書き込む
SEEK	ファイル内の現在参照位置を変更する
DELETE	ファイルを削除する
RENAME	ファイル名、ディレクトリ名を変更する
MKDIR	ディレクトリを作成する
RMDIR	ディレクトリを削除する
OPENDIR	ディレクトリをオープンする
CLOSEDIR	ディレクトリをクローズする
READDIR	ディレクトリからデータを読み込む
SEEKDIR	ディレクトリ内の現在参照位置を変更する
FORMAT	ファイルシステムの論理フォーマットを行う

[図10]

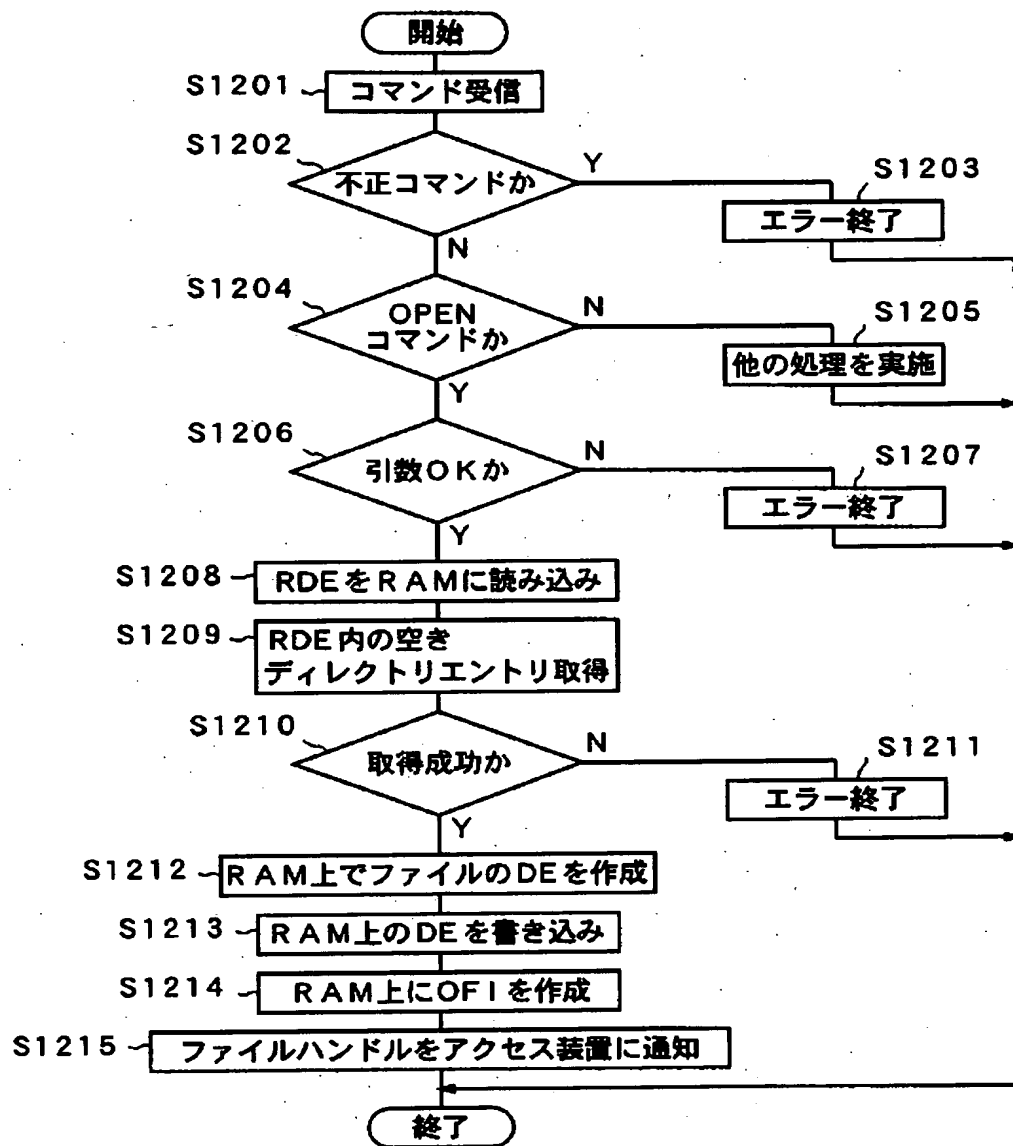




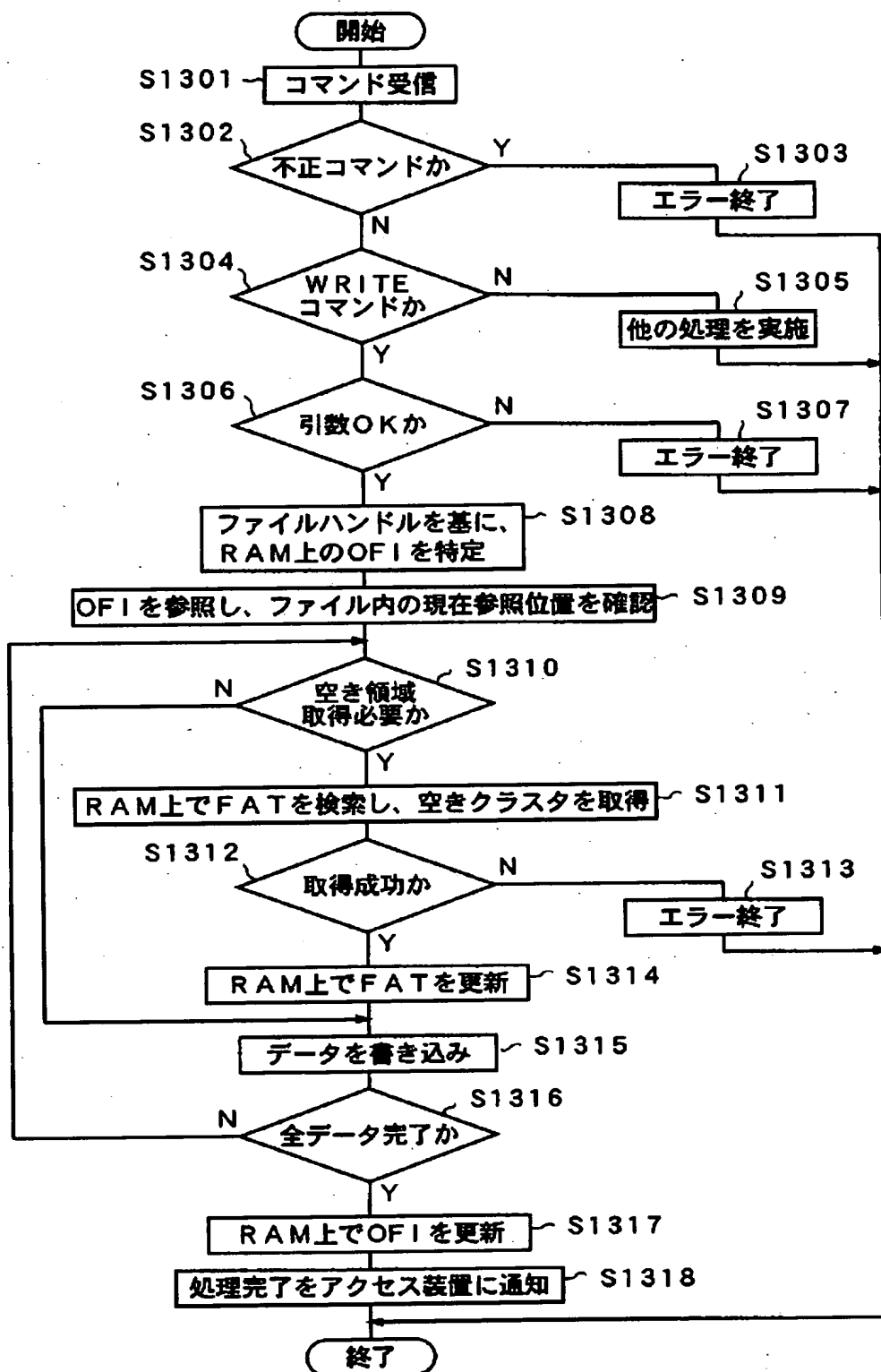
[図11]



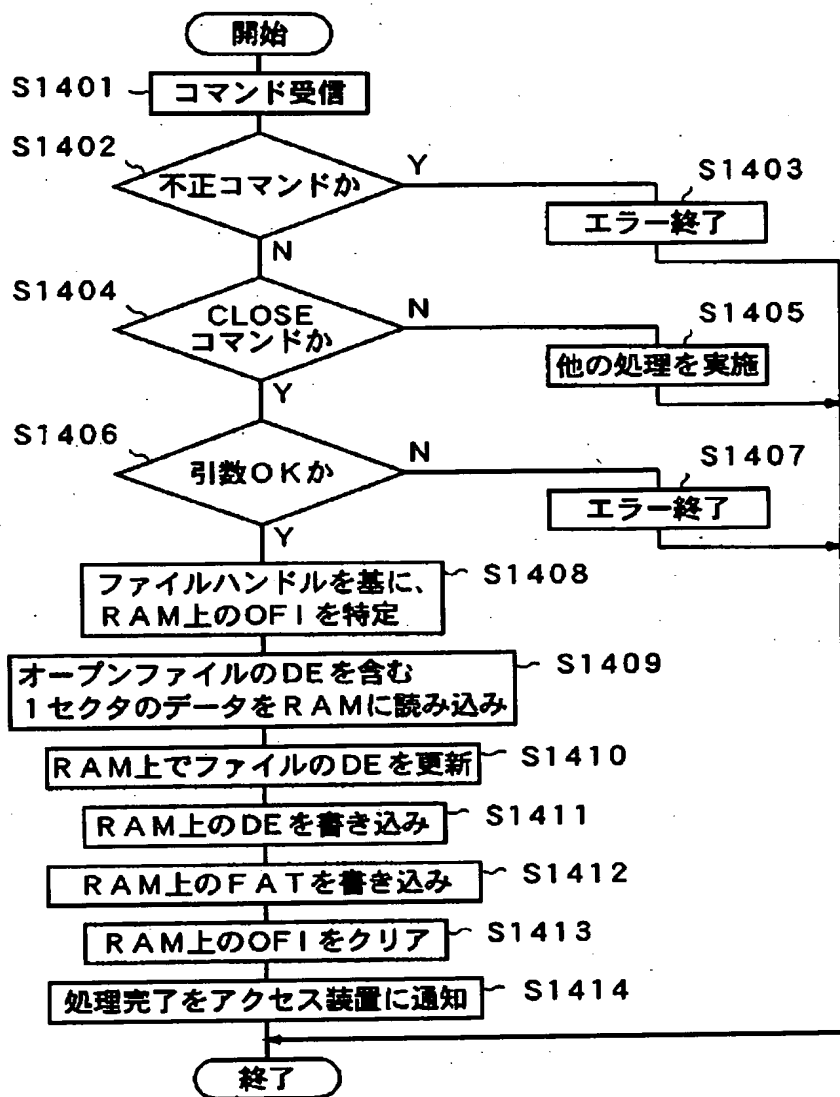
[図12]



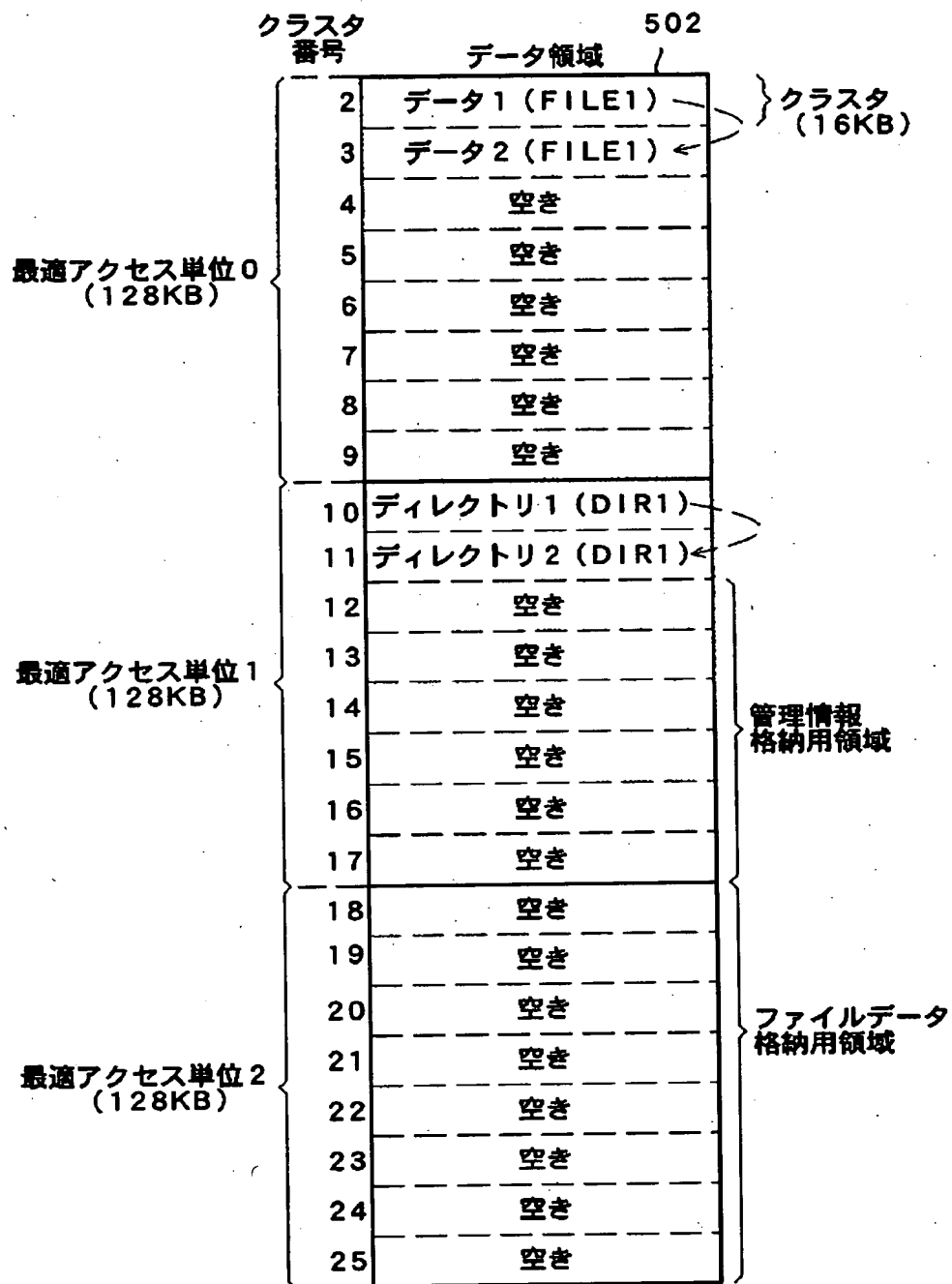
[図13]



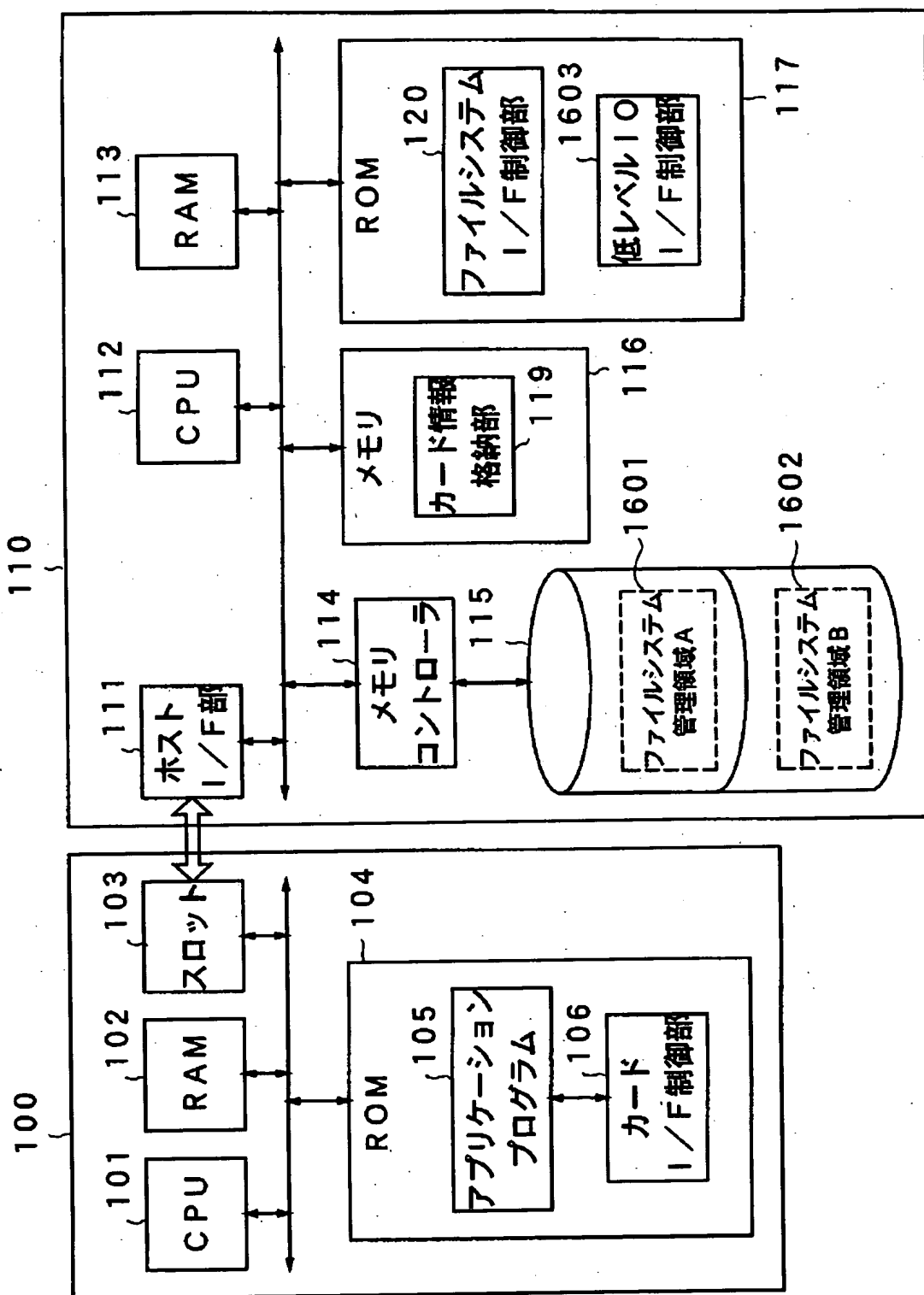
[図14]



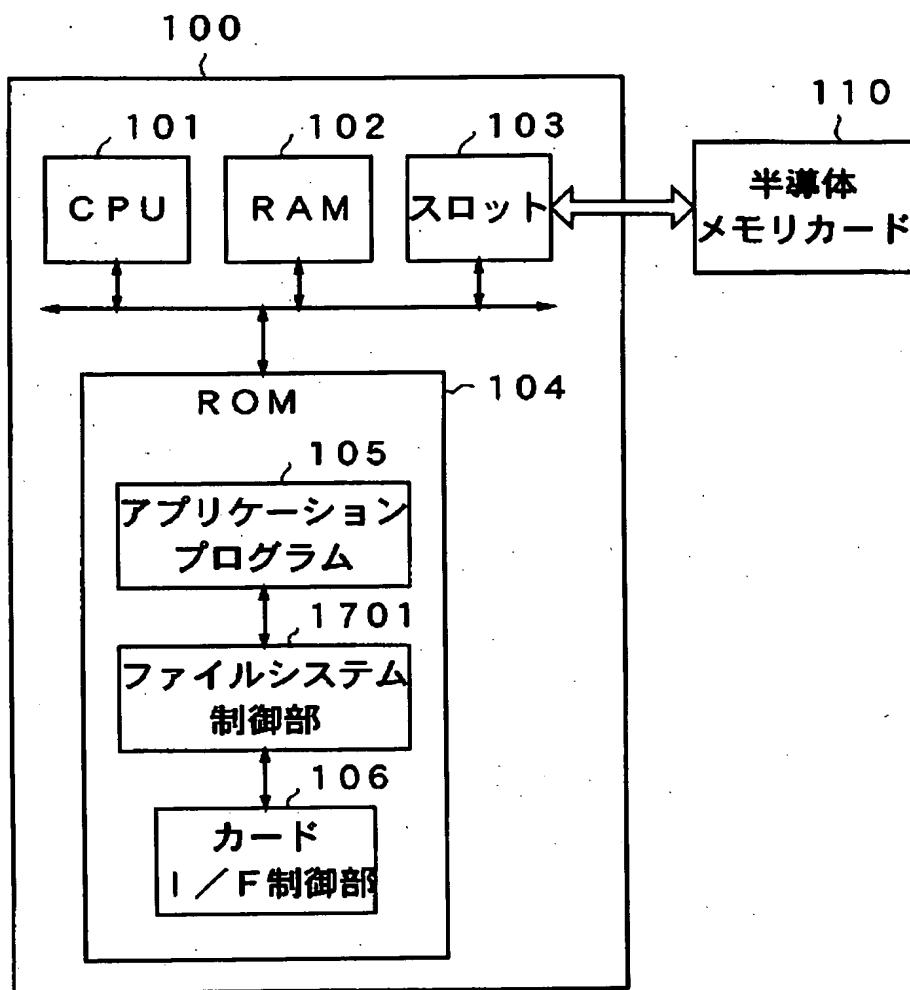
[図15]



[図16]



[図17]

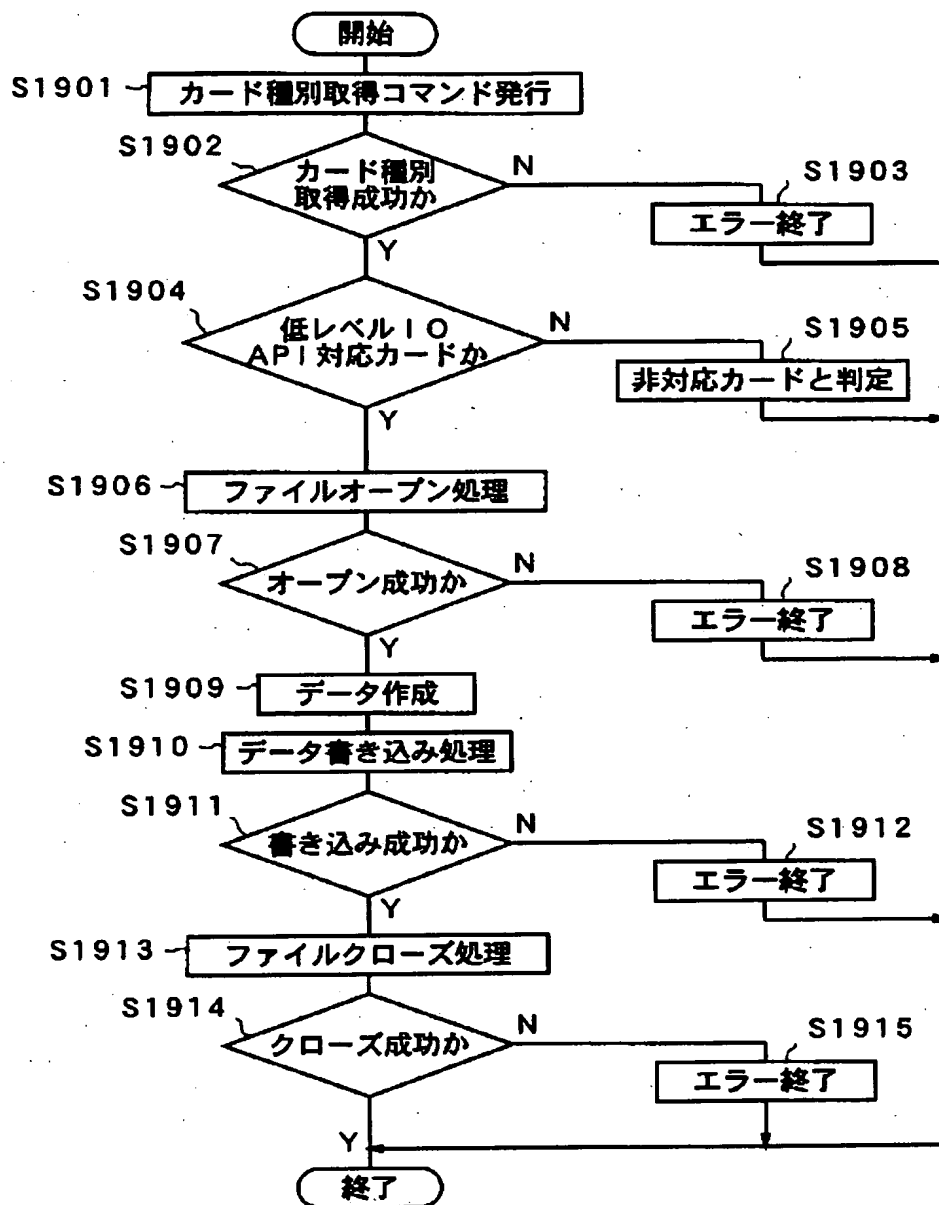


[図18]

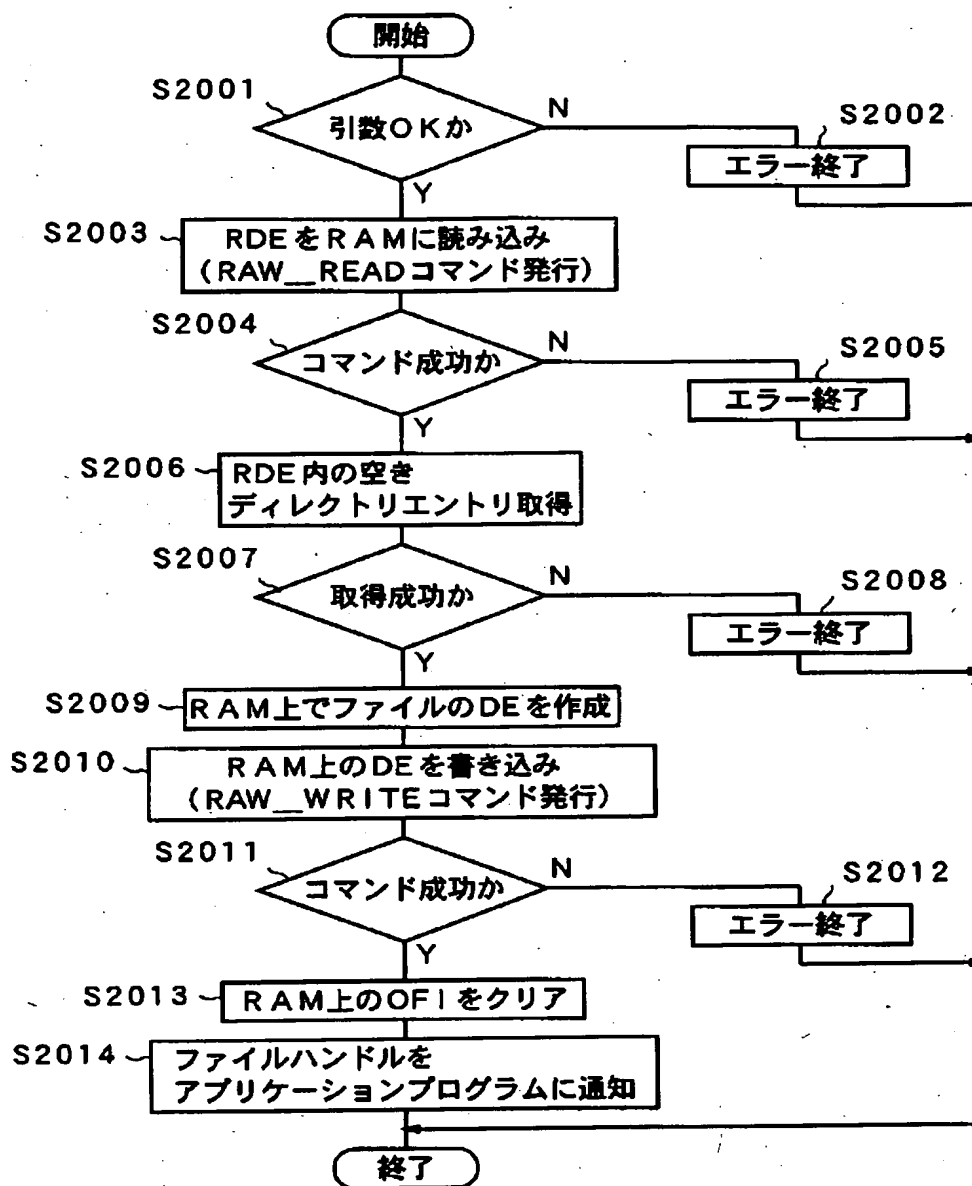
低レベル I O A P I	項目
RAW_READ	データを読み込む
RAW_WRITE	データを書き込む
RAW_ERASE	データを消去する



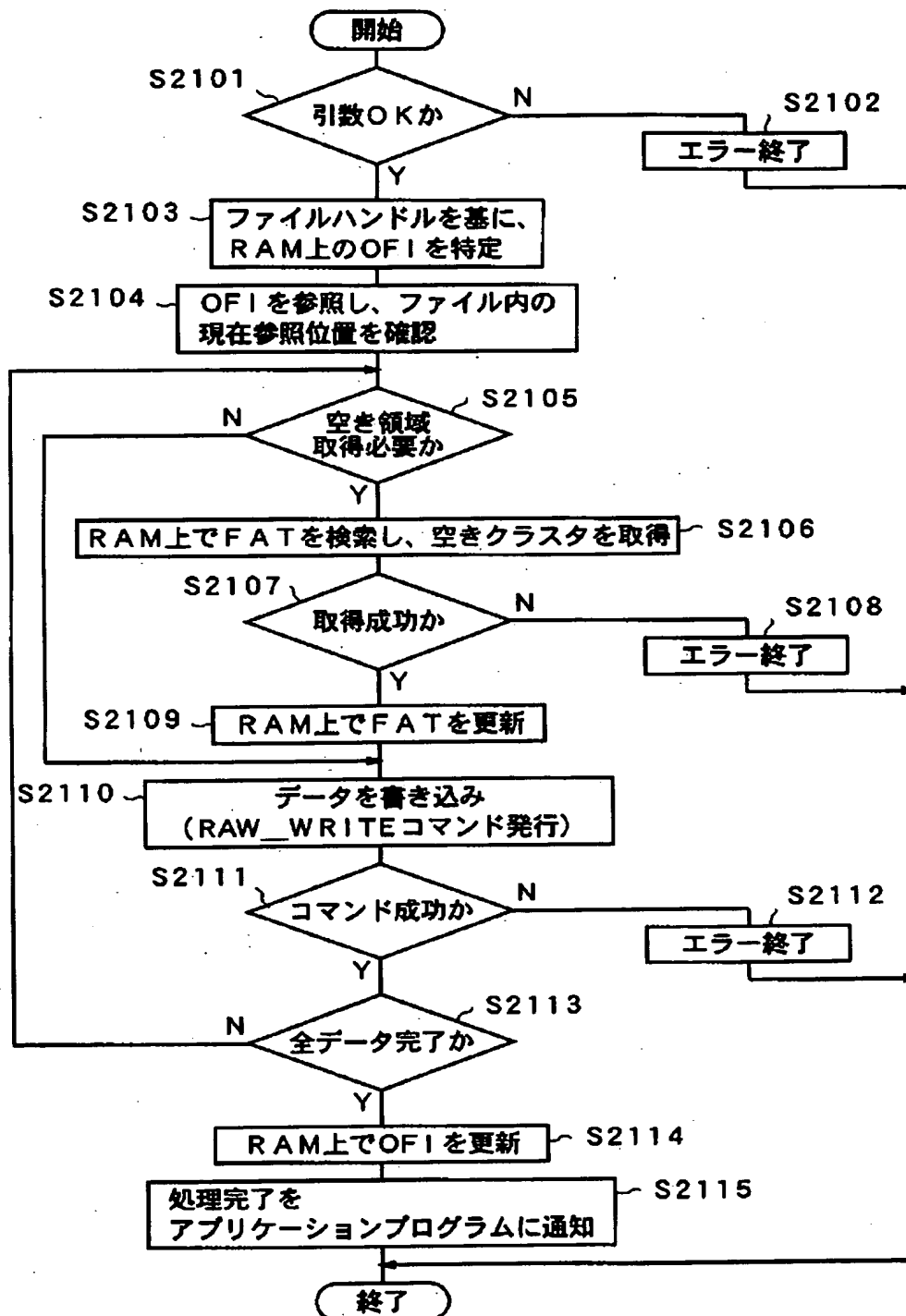
[図19]



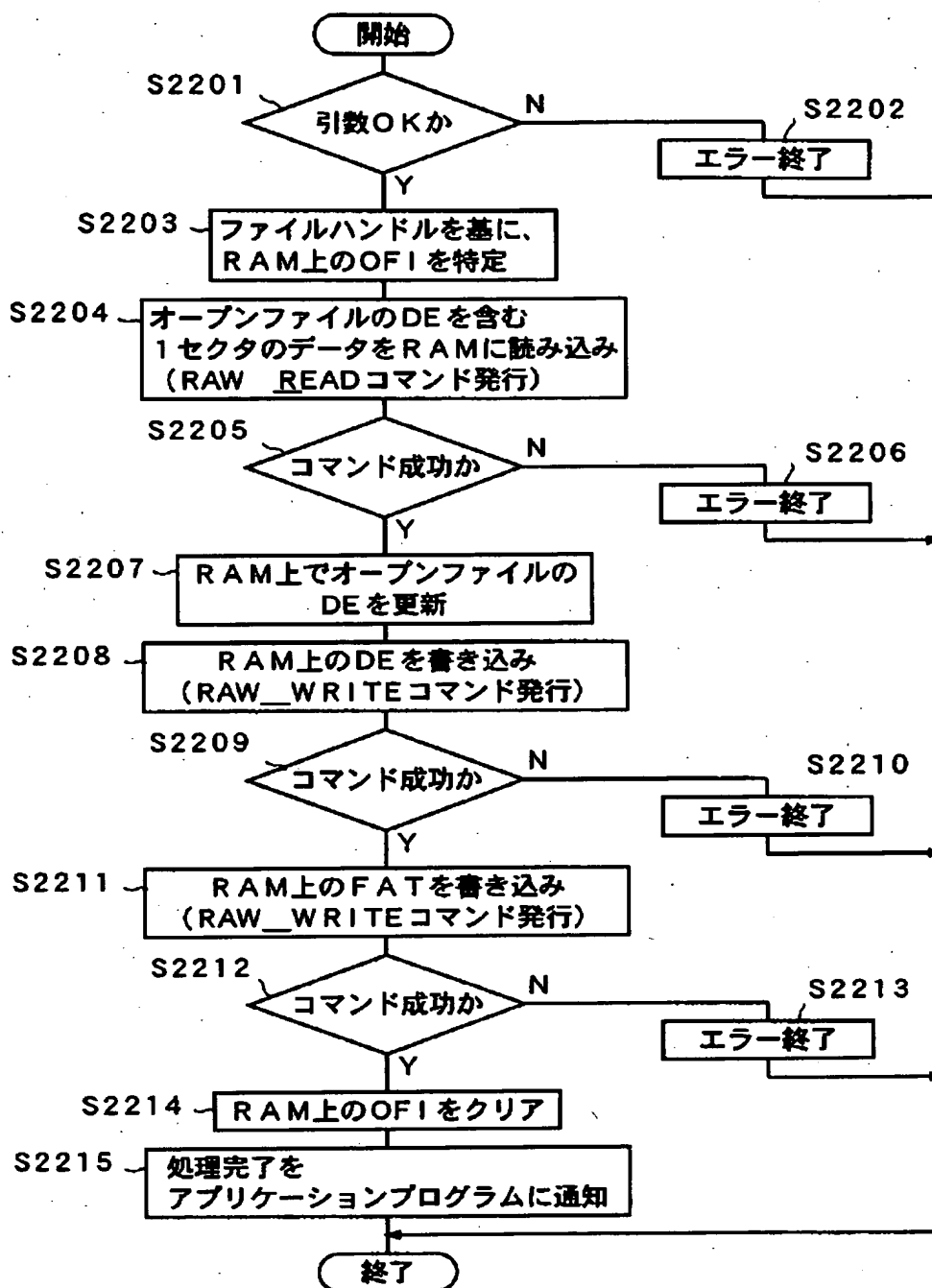
[図20]



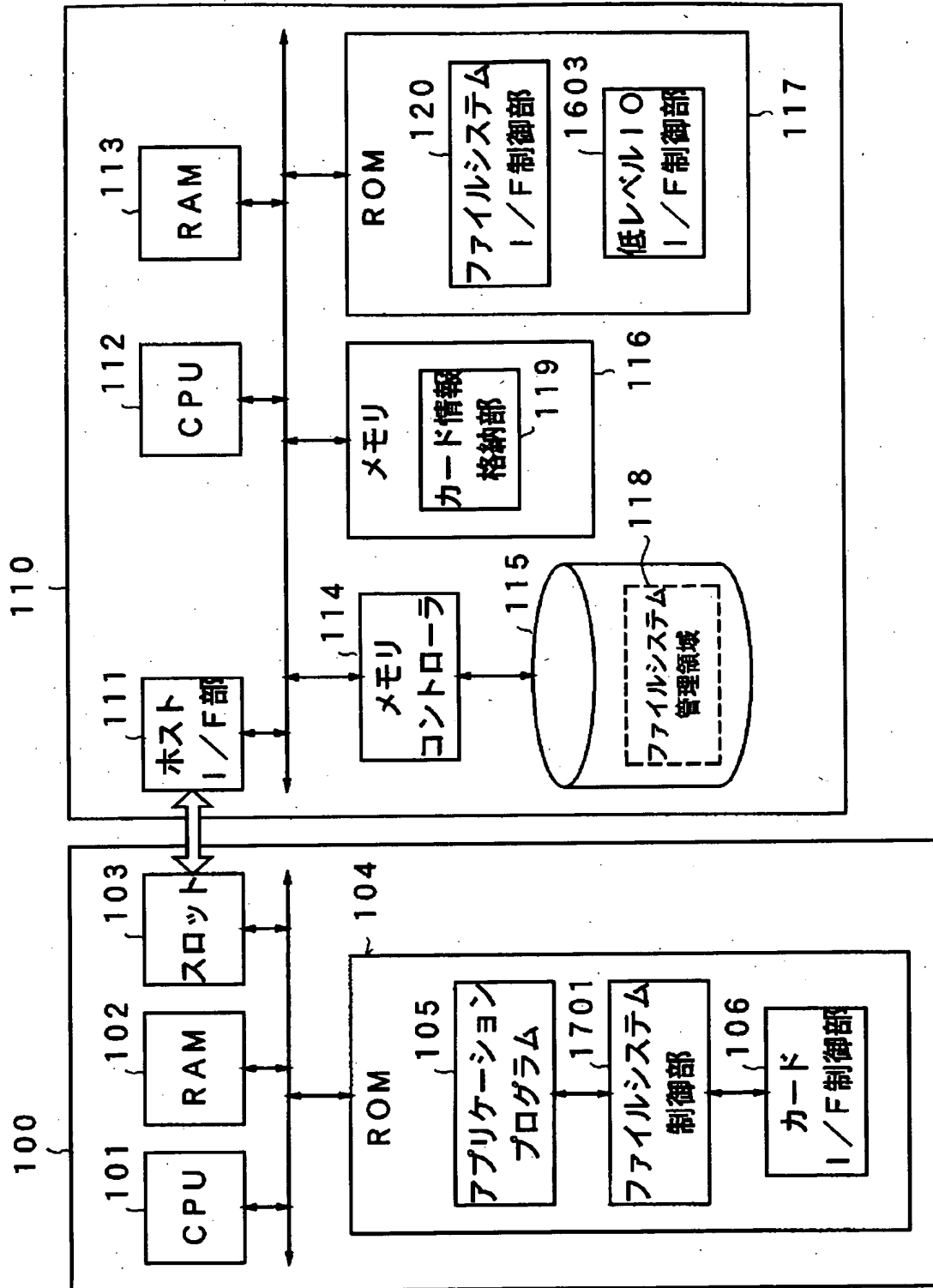
[図21]



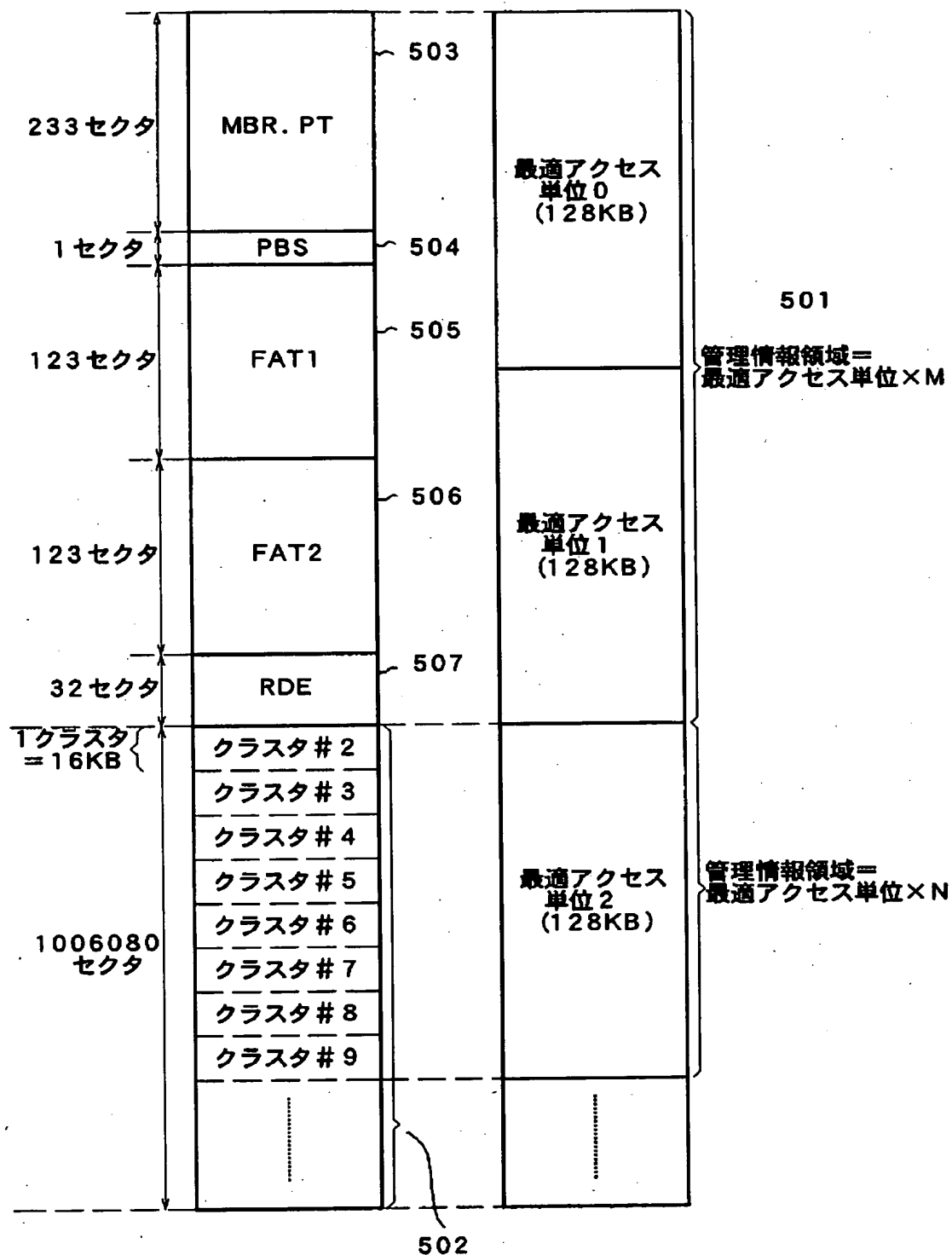
[図22]



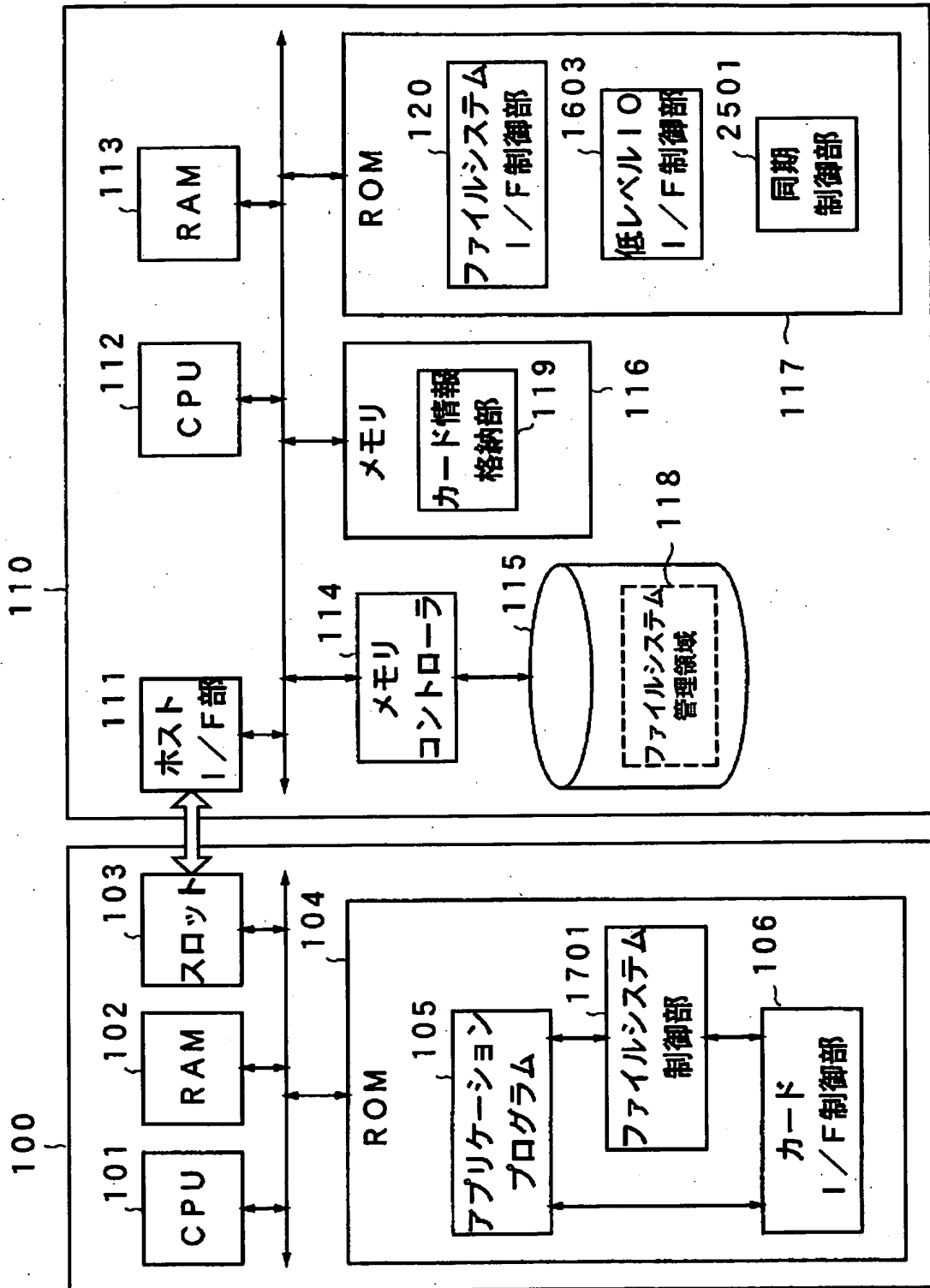
[図23]



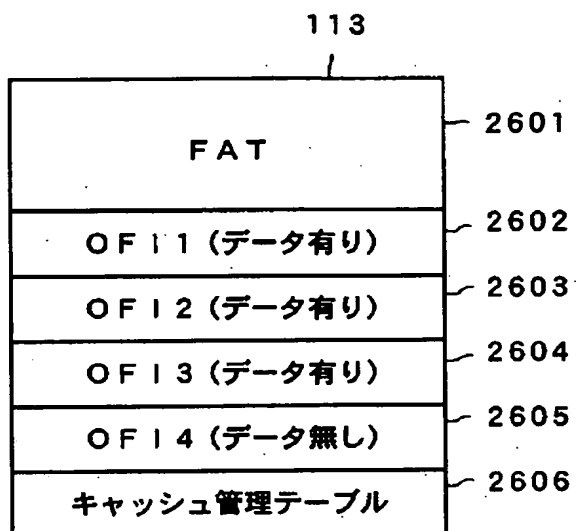
[図24]



[図25]



[図26]



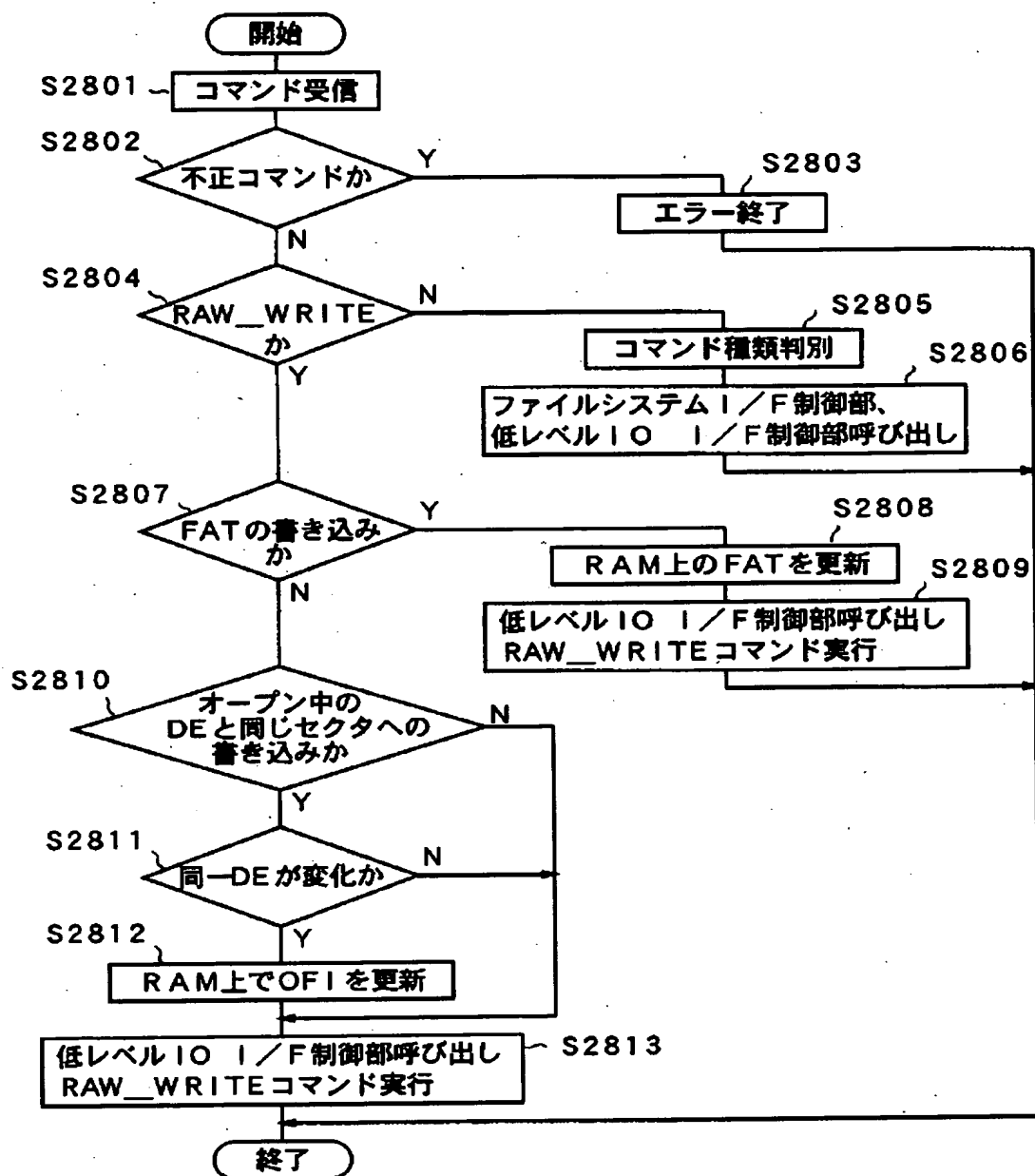
[図27]

2606

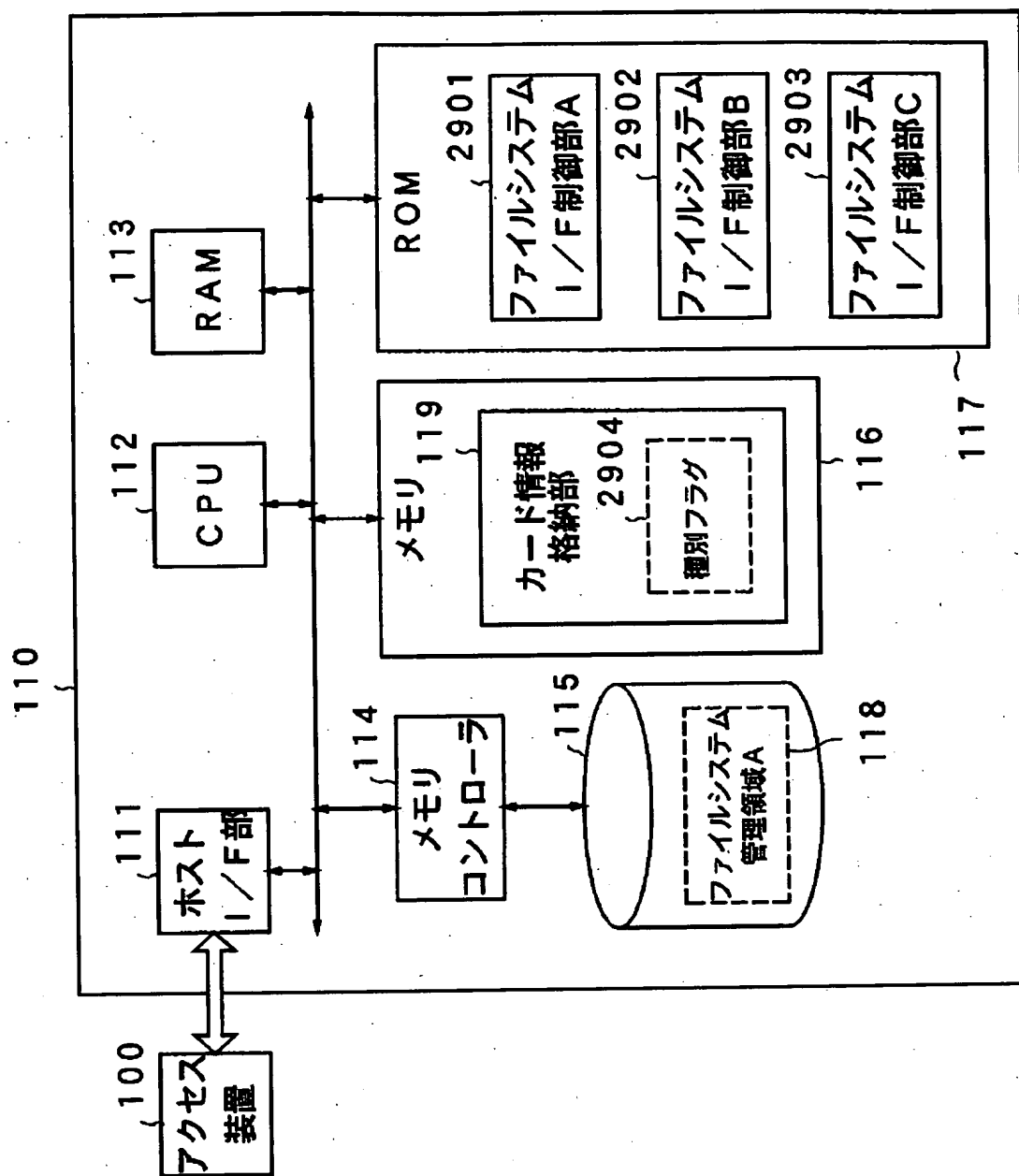
	開始セクタ番号	サイズ
FAT	234セクタ	123セクタ
オープン中のDE 1	480セクタ	1セクタ
オープン中のDE 2	480セクタ	1セクタ
オープン中のDE 3	513セクタ	1セクタ
オープン中のDE 4	0セクタ	0セクタ



[図28]



[図29]



[図30]

